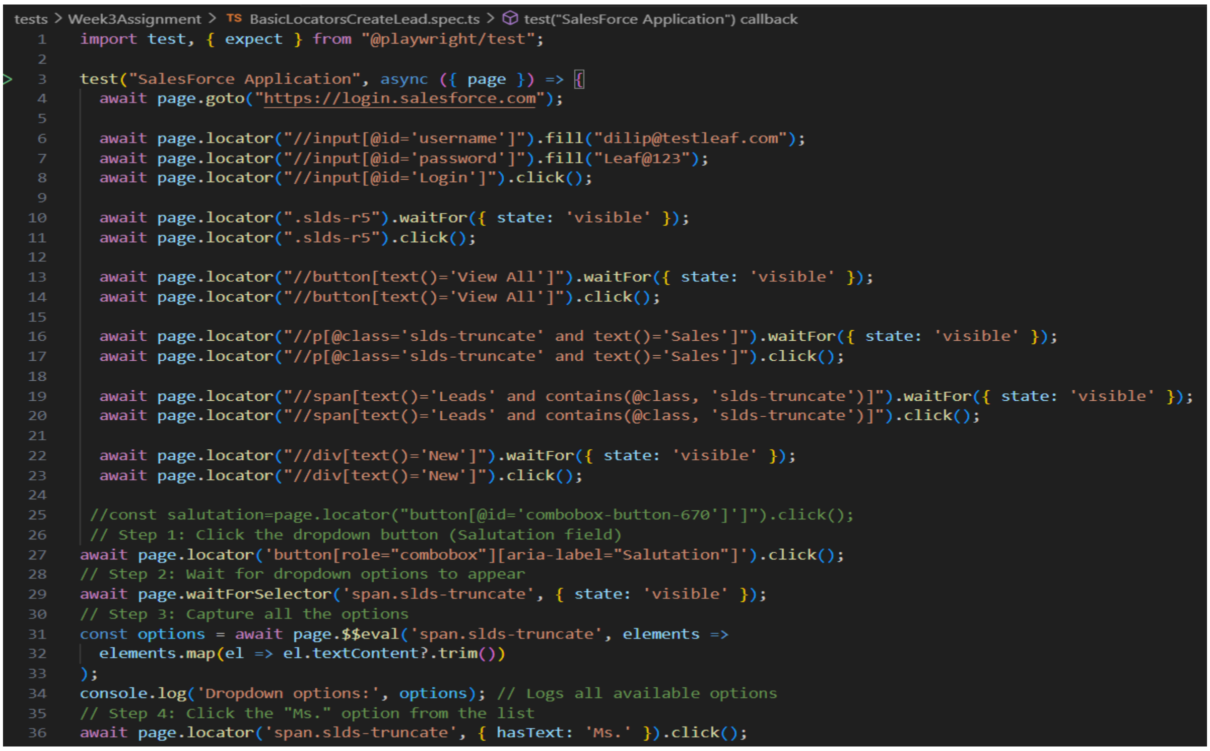
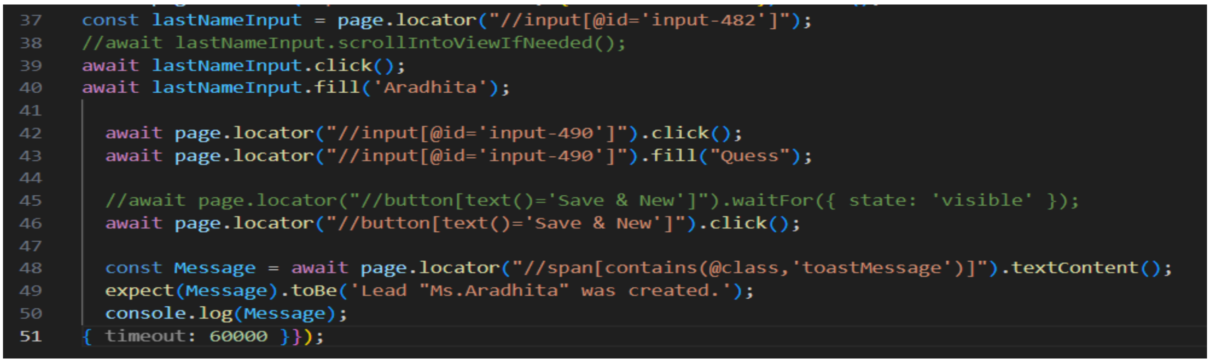
1. Create Lead

```
  import test, { expect } from "@playwright/test";
  test("SalesForce Application", async ({ page }) => {
  await page.goto("https://login.salesforce.com");

  await page.locator("//input[@id='username']").fill("dilip@testleaf.com");
  await page.locator("//input[@id='password']").fill("Leaf@123");
  await page.locator("//input[@id='Login']").click();

  await page.locator(".slds-r5").waitFor({ state: 'visible' });
  await page.locator(".slds-r5").click();

  await page.locator("//button[text()='View All']").waitFor({ state: 'visible' });
  await page.locator("//button[text()='View All']").click();

  await page.locator("//p[@class='slds-truncate' and text()='Sales']").waitFor({ state: 'visible' });
  await page.locator("//p[@class='slds-truncate' and text()='Sales']").click();

  await page.locator("//span[text()='Leads' and contains(@class, 'slds-truncate')]").waitFor({ state:
'visible' });
  await page.locator("//span[text()='Leads' and contains(@class, 'slds-truncate')]").click();

  await page.locator("//div[text()='New']").waitFor({ state: 'visible' });
  await page.locator("//div[text()='New']").click();
  await page.locator('button[role="combobox"][aria-label="Salutation"]').click();
  await page.waitForSelector('span.slds-truncate', { state: 'visible' });

  const options = await page.$$eval('span.slds-truncate', elements =>
  elements.map(el => el.textContent?.trim()));
  console.log('Dropdown options:', options); // Logs all available options
  await page.locator('span.slds-truncate', { hasText: 'Ms.' }).click();

 const lastNameInput = page.locator("//input[@id='input-482']");
 await lastNameInput.scrollIntoViewIfNeeded();
 await lastNameInput.click();
 await lastNameInput.fill('Aradhita');

  await page.locator("//input[@id='input-490']").click();
  await page.locator("//input[@id='input-490']").fill("Quess");
  await page.locator("//button[text()='Save & New']").click();
  const Message = await page.locator("//span[contains(@class,'toastMessage')]").textContent();
  expect(Message).toBe('Lead "Ms.Aradhita" was created.');
  console.log(Message);
{ timeout: 60000 }});
```



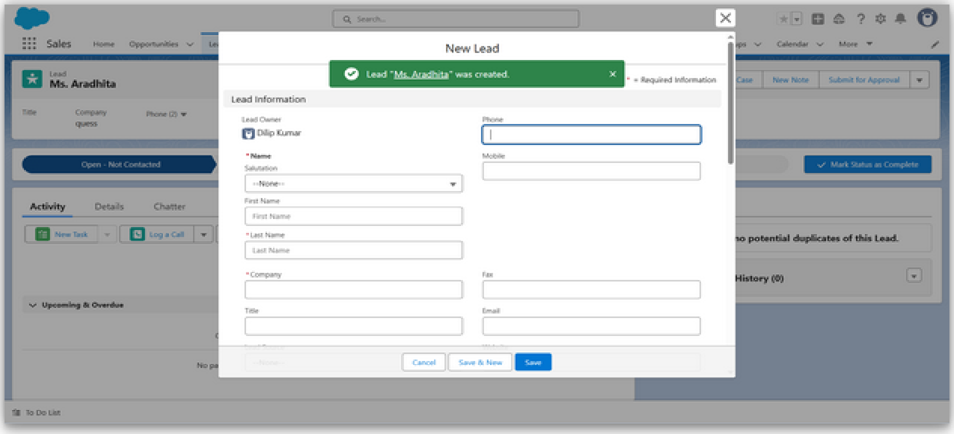
tests > Week3Assignment > TS BasicLocatorsCreateLead.spec.ts > test("SalesForce Application") callback

```
import test, { expect } from "@playwright/test";

test("SalesForce Application", async ({ page }) => {
  await page.goto("https://login.salesforce.com");

  await page.locator("//input[@id='username']").fill("dilip@testleaf.com");
  await page.locator("//input[@id='password']").fill("Leaf@123");
  await page.locator("//input[@id='Login']").click();

  await page.locator(".slds-r5").waitFor({ state: 'visible' });
  await page.locator(".slds-r5").click();

  await page.locator("//button[text()='View All']").waitFor({ state: 'visible' });
  await page.locator("//button[text()='View All']").click();

  await page.locator("//p[@class='slds-truncate' and text()='Sales']").waitFor({ state: 'visible' });
  await page.locator("//p[@class='slds-truncate' and text()='Sales']").click();

  await page.locator("//span[text()='Leads' and contains(@class, 'slds-truncate')]").waitFor({ state: 'visible' });
  await page.locator("//span[text()='Leads' and contains(@class, 'slds-truncate')]").click();

  await page.locator("//div[text()='New']").waitFor({ state: 'visible' });
  await page.locator("//div[text()='New']").click();

 //const salutation=page.locator("button[@id='combobox-button-670']']").click();
 // Step 1: Click the dropdown button (Salutation field)
await page.locator('button[role="combobox"][aria-label="Salutation"]').click();
// Step 2: Wait for dropdown options to appear
await page.waitForSelector('span.slds-truncate', { state: 'visible' });
// Step 3: Capture all the options
const options = await page.$$eval('span.slds-truncate', elements =>
  elements.map(el => el.textContent?.trim())
);
console.log('Dropdown options:', options); // Logs all available options
// Step 4: Click the "Ms." option from the list
await page.locator('span.slds-truncate', { hasText: 'Ms.' }).click();
```



```
const lastNameInput = page.locator("//input[@id='input-482']");
//await lastNameInput.scrollIntoViewIfNeeded();
await lastNameInput.click();
await lastNameInput.fill('Aradhita');

  await page.locator("//input[@id='input-490']").click();
  await page.locator("//input[@id='input-490']").fill("Quess");

  //await page.locator("//button[text()='Save & New']").waitFor({ state: 'visible' });
  await page.locator("//button[text()='Save & New']").click();

  const Message = await page.locator("//span[contains(@class,'toastMessage')]").textContent();
  expect(Message).toBe('Lead "Ms.Aradhita" was created.');
  console.log(Message);
{ timeout: 60000 }});
```

Screenshots

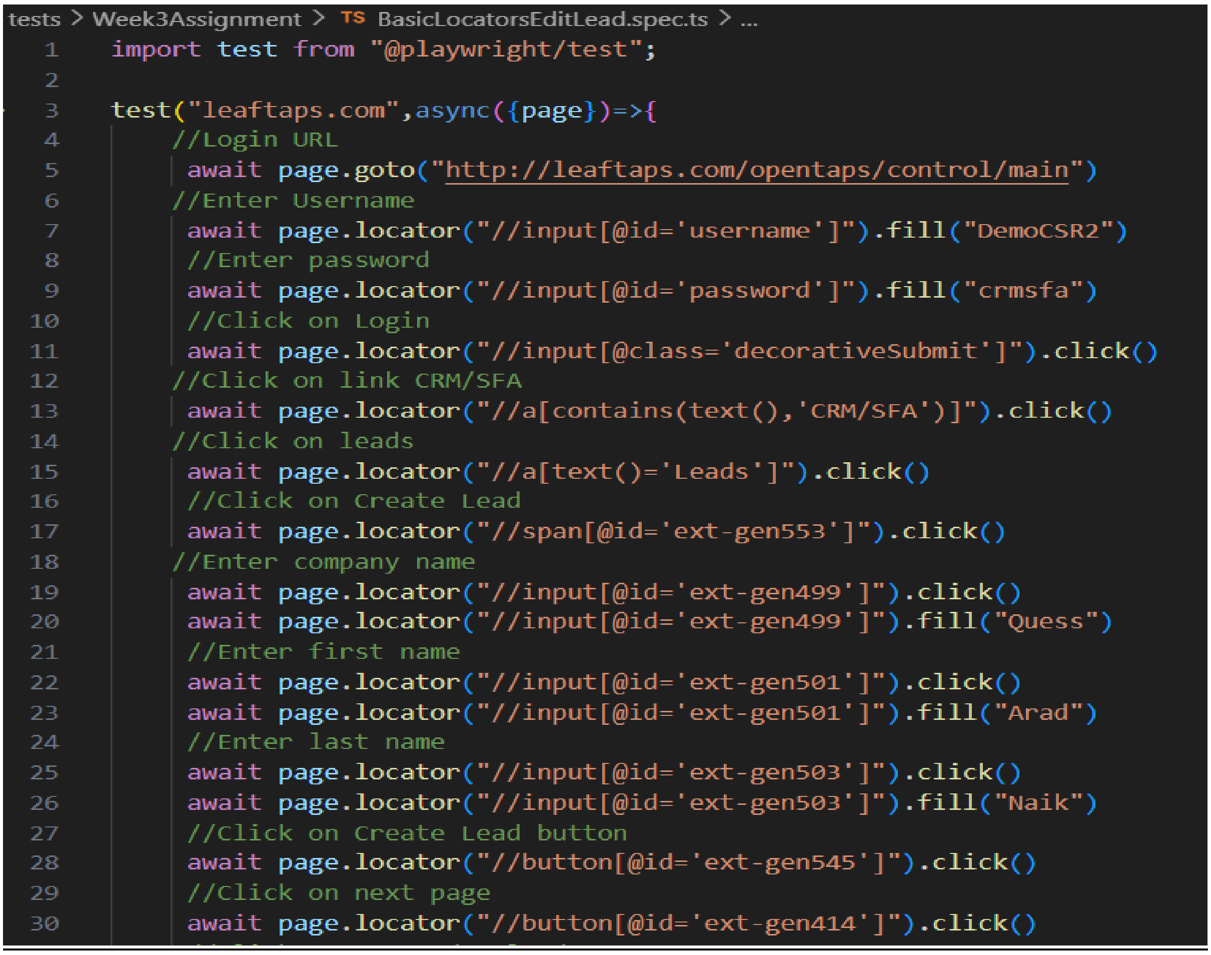


Lead "Ms. Aradhita" was created.

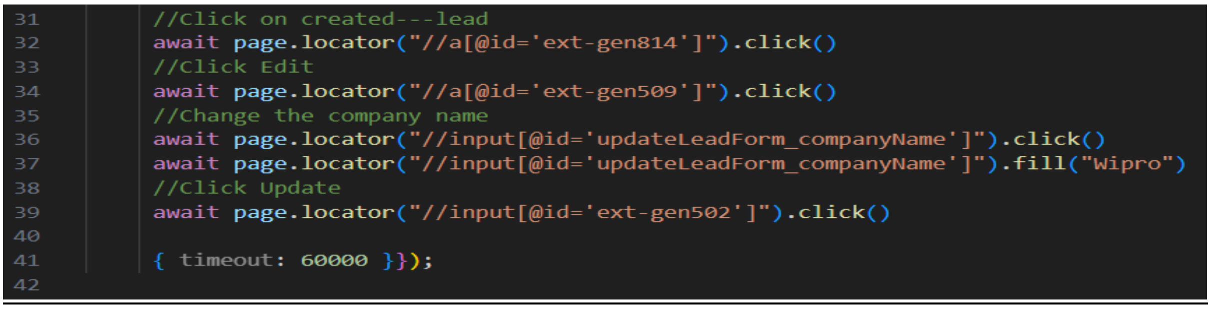
screenshot

2. Edit Lead

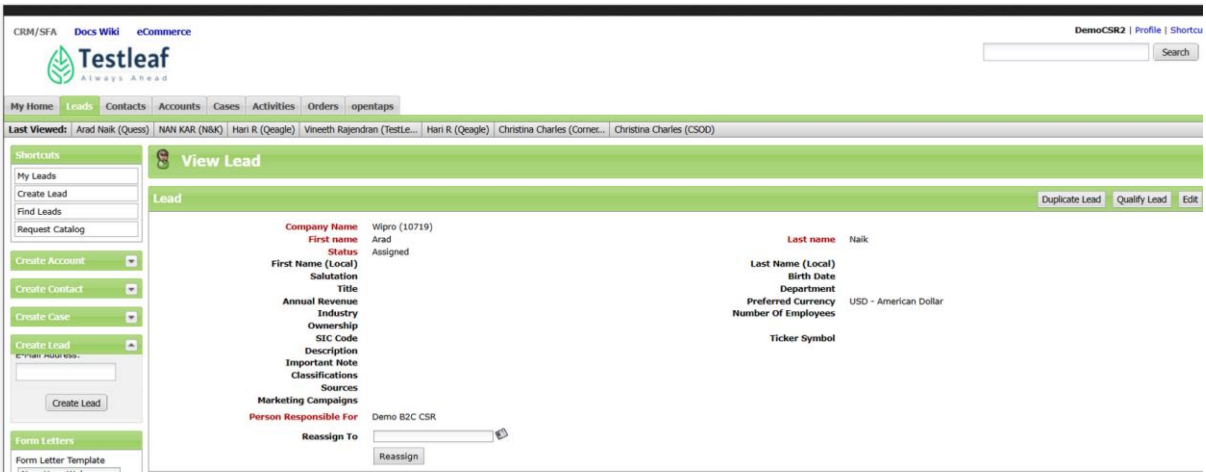
```
import test from "@playwright/test";

test("leaftaps.com",async({page})=>{
  //Login URL
   await page.goto("http://leaftaps.com/opentaps/control/main")
  //Enter Username
   await page.locator("//input[@id='username']").fill("DemoCSR2")
   //Enter password
   await page.locator("//input[@id='password']").fill("crmsfa")
   //Click on Login
   await page.locator("//input[@class='decorativeSubmit']").click()
  //Click on link CRM/SFA
   await page.locator("//a[contains(text(),'CRM/SFA')]").click()
  //Click on leads
   await page.locator("//a[text()='Leads']").click()
   //Click on Create Lead
   await page.locator("//span[@id='ext-gen553']").click()
  //Enter company name
   await page.locator("//input[@id='ext-gen499']").click()
   await page.locator("//input[@id='ext-gen499']").fill("Quess")
   //Enter first name
   await page.locator("//input[@id='ext-gen501']").click()
   await page.locator("//input[@id='ext-gen501']").fill("Arad")
   //Enter last name
   await page.locator("//input[@id='ext-gen503']").click()
   await page.locator("//input[@id='ext-gen503']").fill("Naik")
   //Click on Create Lead button
   await page.locator("//button[@id='ext-gen545']").click()
   //Click on next page
   await page.locator("//button[@id='ext-gen414']").click()
   //Click on created---lead
   await page.locator("//a[@id='ext-gen814']").click()
   //Click Edit
   await page.locator("//a[@id='ext-gen509']").click()
   //Change the company name
   await page.locator("//input[@id='updateLeadForm_companyName']").click()
   await page.locator("//input[@id='updateLeadForm_companyName']").fill("Wipro")
   //Click Update
   await page.locator("//input[@id='ext-gen502']").click()
   { timeout: 60000 }});
```


tests > Week3Assignment > TS BasicLocatorsEditLead.spec.ts > ...

```
import test from "@playwright/test";

test("leaftaps.com",async({page})=>{
    //Login URL
     await page.goto("http://leaftaps.com/opentaps/control/main")
    //Enter Username
     await page.locator("//input[@id='username']").fill("DemoCSR2")
     //Enter password
     await page.locator("//input[@id='password']").fill("crmsfa")
     //Click on Login
     await page.locator("//input[@class='decorativeSubmit']").click()
    //Click on link CRM/SFA
     await page.locator("//a[contains(text(),'CRM/SFA')]").click()
    //Click on leads
     await page.locator("//a[text()='Leads']").click()
     //Click on Create Lead
     await page.locator("//span[@id='ext-gen553']").click()
    //Enter company name
     await page.locator("//input[@id='ext-gen499']").click()
     await page.locator("//input[@id='ext-gen499']").fill("Quess")
     //Enter first name
     await page.locator("//input[@id='ext-gen501']").click()
     await page.locator("//input[@id='ext-gen501']").fill("Arad")
     //Enter last name
     await page.locator("//input[@id='ext-gen503']").click()
     await page.locator("//input[@id='ext-gen503']").fill("Naik")
     //Click on Create Lead button
     await page.locator("//button[@id='ext-gen545']").click()
     //Click on next page
     await page.locator("//button[@id='ext-gen414']").click()
```



```
    //Click on created---lead
    await page.locator("//a[@id='ext-gen814']").click()
    //Click Edit
    await page.locator("//a[@id='ext-gen509']").click()
    //Change the company name
    await page.locator("//input[@id='updateLeadForm_companyName']").click()
    await page.locator("//input[@id='updateLeadForm_companyName']").fill("Wipro")
    //Click Update
    await page.locator("//input[@id='ext-gen502']").click()

    { timeout: 60000 }});

```



Last Viewed: Arad Naik (Quess) | NAN KAR (N&K) | Hari R (Qeagle) | Vineeth Rajendran (TestLe... | Hari R (Qeagle) | Christina Charles (Corner... | Christina Charles (CSOD)

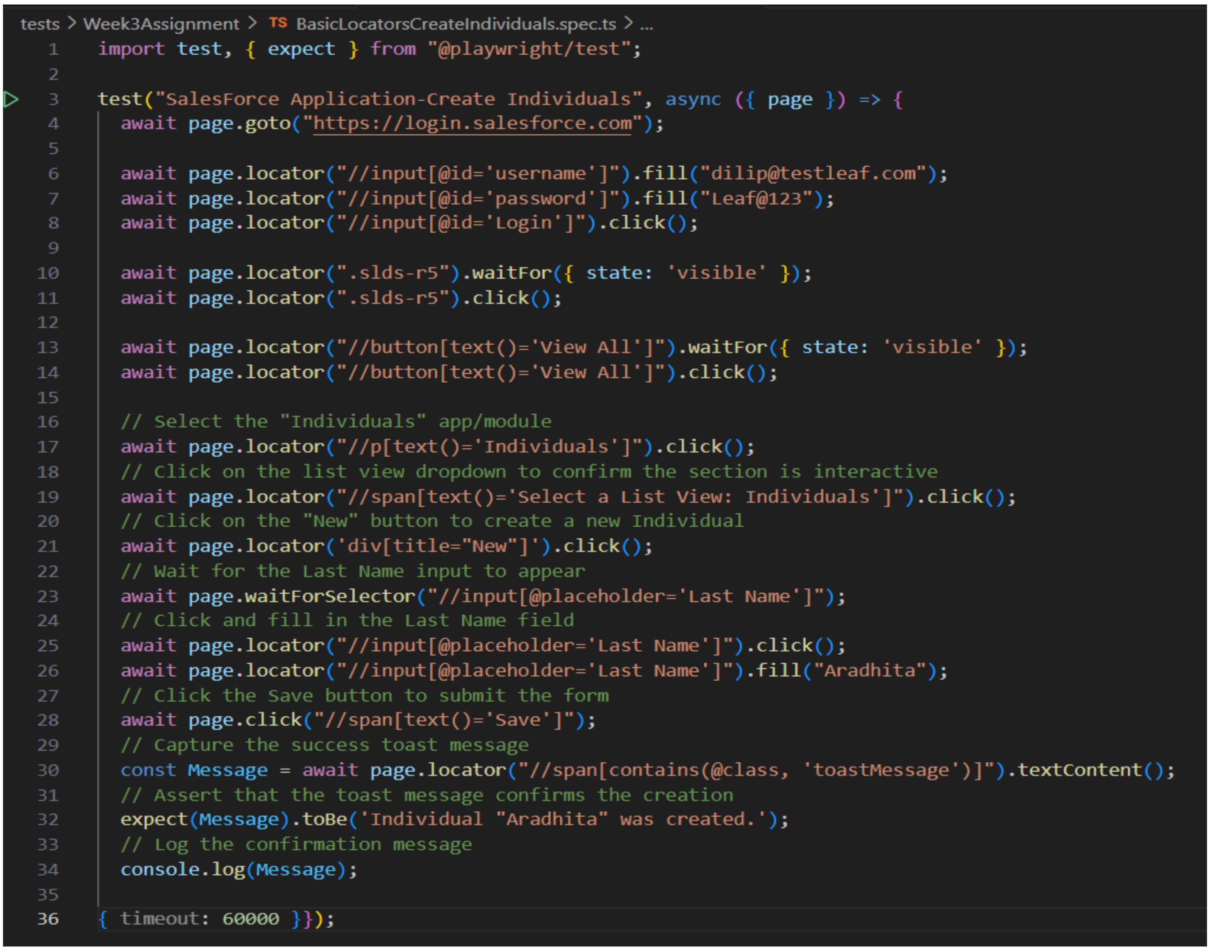
View Lead

Lead

Company Name	Wipro (10719)		
First name	Arad	Last name	Naik
Status	Assigned		
First Name (Local)		Last Name (Local)	
Salutation		Birth Date	
Title		Department	
Annual Revenue		Preferred Currency	USD - American Dollar
Industry		Number Of Employees	
Ownership			
SIC Code		Ticker Symbol	
Description			
Important Note			
Classifications			
Sources			
Marketing Campaigns			
Person Responsible For	Demo B2C CSR		
Reassign To			

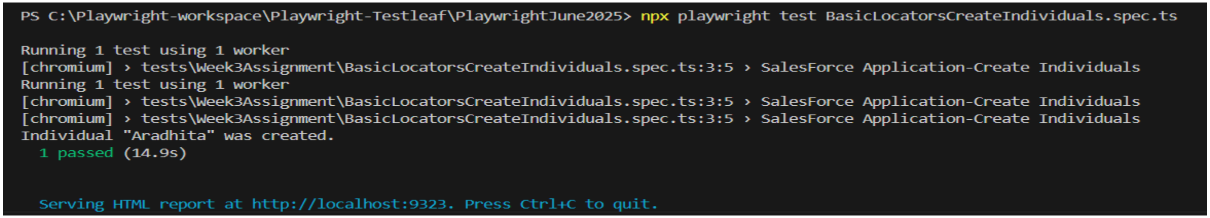
3. Create Individuals

```
import test, { expect } from "@playwright/test";

test("SalesForce Application-Create Individuals", async ({ page }) => {
  await page.goto("https://login.salesforce.com");

  await page.locator("//input[@id='username']").fill("dilip@testleaf.com");
  await page.locator("//input[@id='password']").fill("Leaf@123");
  await page.locator("//input[@id='Login']").click();

  await page.locator(".slds-r5").waitFor({ state: 'visible' });
  await page.locator(".slds-r5").click();

  await page.locator("//button[text()='View All']").waitFor({ state: 'visible' });
  await page.locator("//button[text()='View All']").click();

  // Select the "Individuals" app/module
  await page.locator("//p[text()='Individuals']").click();
  // Click on the list view dropdown to confirm the section is interactive
  await page.locator("//span[text()='Select a List View: Individuals']").click();
  // Click on the "New" button to create a new Individual
  await page.locator('div[title="New"]').click();
  // Wait for the Last Name input to appear
  await page.waitForSelector("//input[@placeholder='Last Name']");
  // Click and fill in the Last Name field
  await page.locator("//input[@placeholder='Last Name']").click();
  await page.locator("//input[@placeholder='Last Name']").fill("Aradhita");
  // Click the Save button to submit the form
  await page.click("//span[text()='Save']");
  // Capture the success toast message
  const Message = await page.locator("//span[contains(@class, 'toastMessage')]").textContent();
  // Assert that the toast message confirms the creation
  expect(Message).toBe('Individual "Aradhita" was created.');
  // Log the confirmation message
  console.log(Message);

{ timeout: 60000 }});
```

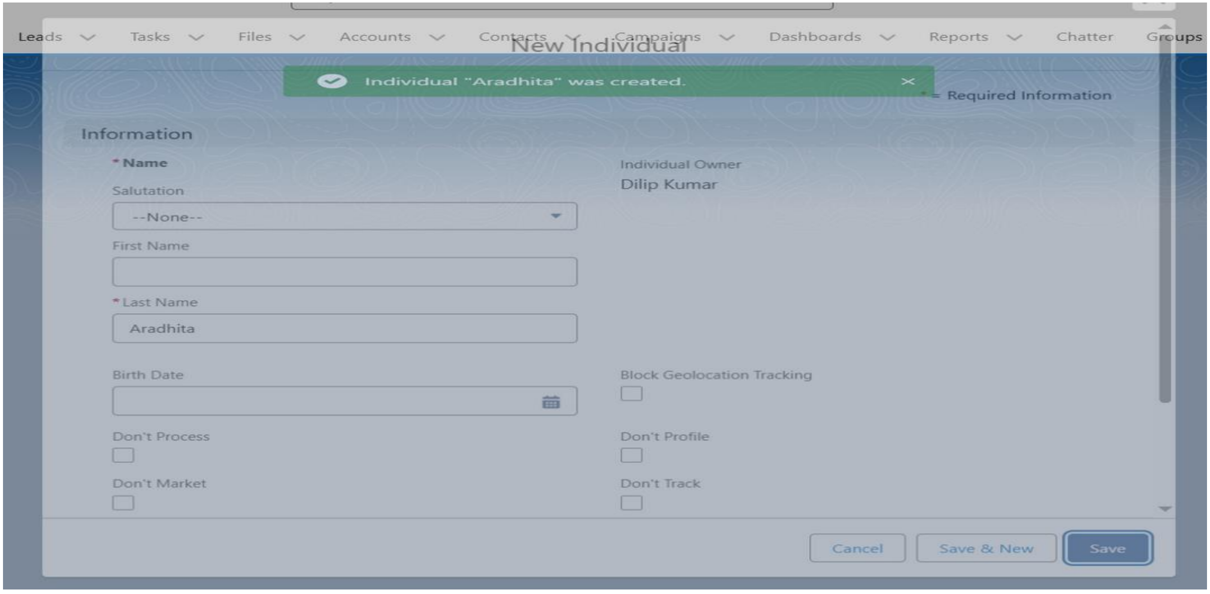


tests > Week3Assignment > TS BasicLocatorsCreateIndividuals.spec.ts > ...

```ts
import test, { expect } from "@playwright/test";

test("SalesForce Application-Create Individuals", async ({ page }) => {
  await page.goto("https://login.salesforce.com");

  await page.locator("//input[@id='username']").fill("dilip@testleaf.com");
  await page.locator("//input[@id='password']").fill("Leaf@123");
  await page.locator("//input[@id='Login']").click();

  await page.locator(".slds-r5").waitFor({ state: 'visible' });
  await page.locator(".slds-r5").click();

  await page.locator("//button[text()='View All']").waitFor({ state: 'visible' });
  await page.locator("//button[text()='View All']").click();

  // Select the "Individuals" app/module
  await page.locator("//p[text()='Individuals']").click();
  // Click on the list view dropdown to confirm the section is interactive
  await page.locator("//span[text()='Select a List View: Individuals']").click();
  // Click on the "New" button to create a new Individual
  await page.locator('div[title="New"]').click();
  // Wait for the Last Name input to appear
  await page.waitForSelector("//input[@placeholder='Last Name']");
  // Click and fill in the Last Name field
  await page.locator("//input[@placeholder='Last Name']").click();
  await page.locator("//input[@placeholder='Last Name']").fill("Aradhita");
  // Click the Save button to submit the form
  await page.click("//span[text()='Save']");
  // Capture the success toast message
  const Message = await page.locator("//span[contains(@class, 'toastMessage')]").textContent();
  // Assert that the toast message confirms the creation
  expect(Message).toBe('Individual "Aradhita" was created.');
  // Log the confirmation message
  console.log(Message);

{ timeout: 60000 }});
```



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

Running 1 test using 1 worker
[chromium] › tests\Week3Assignment\BasicLocatorsCreateIndividuals.spec.ts:3:5 › SalesForce Application-Create Individuals
Running 1 test using 1 worker
[chromium] › tests\Week3Assignment\BasicLocatorsCreateIndividuals.spec.ts:3:5 › SalesForce Application-Create Individuals
[chromium] › tests\Week3Assignment\BasicLocatorsCreateIndividuals.spec.ts:3:5 › SalesForce Application-Create Individuals
Individual "Aradhita" was created.
  1 passed (14.9s)

  Serving HTML report at http://localhost:9323. Press Ctrl+C to quit.
```



New Individual

Individual "Aradhita" was created.

* = Required Information

Information

* Name

Salutation

--None--

First Name

* Last Name

Aradhita

Birth Date

Don't Process □

Don't Market □

Individual Owner

Dilip Kumar

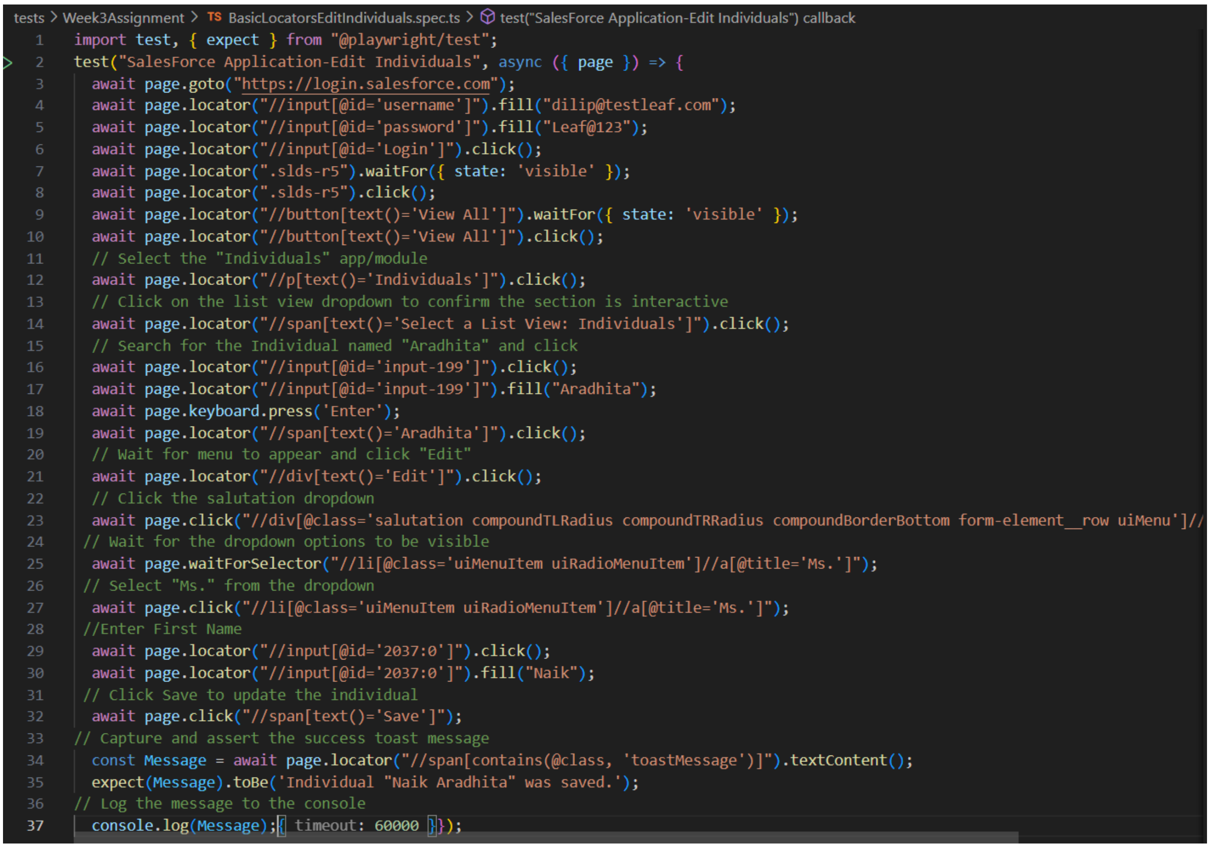
Block Geolocation Tracking □

Don't Profile □

Don't Track □

4. Edit Individuals

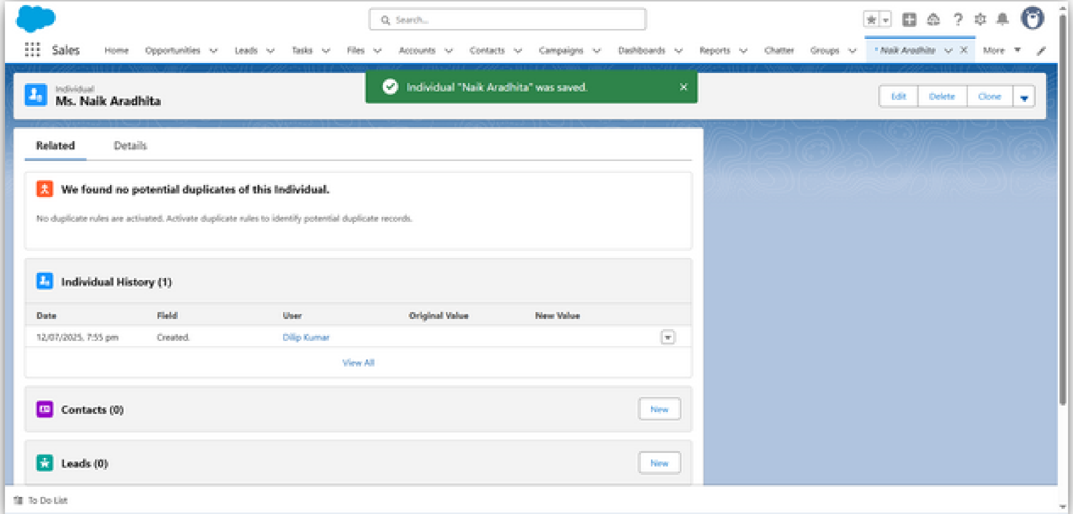
```
import test, { expect } from "@playwright/test";

 test("SalesForce Application-Edit Individuals", async ({ page }) => {
 await page.goto("https://login.salesforce.com");
 await page.locator("//input[@id='username']").fill("dilip@testleaf.com");
 await page.locator("//input[@id='password']").fill("Leaf@123");
 await page.locator("//input[@id='Login']").click();
 await page.locator(".slds-r5").waitFor({ state: 'visible' });
 await page.locator(".slds-r5").click();
 await page.locator("//button[text()='View All']").waitFor({ state: 'visible' });
 await page.locator("//button[text()='View All']").click();
  // Select the "Individuals" app/module
 await page.locator("//p[text()='Individuals']").click();
 // Click on the list view dropdown to confirm the section is interactive
 await page.locator("//span[text()='Select a List View: Individuals']").click();
 // Search for the Individual named "Aradhita" and click
 await page.locator("//input[@id='input-199']").click();
 await page.locator("//input[@id='input-199']").fill("Aradhita");
 await page.keyboard.press('Enter');
 await page.locator("//span[text()='Aradhita']").click();
 // Wait for menu to appear and click "Edit"
  await page.locator("//div[text()='Edit']").click();
 // Click the salutation dropdown
 await page.click("//div[@class='salutation compoundTLRadius compoundTRRadius
compoundBorderBottom form-element__row uiMenu']//a[@class='select']");
 // Wait for the dropdown options to be visible
 await page.waitForSelector("//li[@class='uiMenuItem uiRadioMenuItem']//a[@title='Ms.']");
// Select "Ms." from the dropdown
 await page.click("//li[@class='uiMenuItem uiRadioMenuItem']//a[@title='Ms.']");
 //Enter First Name
 await page.locator("//input[@id='2037:0']").click();
 await page.locator("//input[@id='2037:0']").fill("Naik");
// Click Save to update the individual
 await page.click("//span[text()='Save']");
// Capture and assert the success toast message
 const Message = await page.locator("//span[contains(@class, 'toastMessage')]").textContent();
 expect(Message).toBe('Individual "Naik Aradhita" was saved.');
// Log the message to the console
 console.log(Message);
 { timeout: 60000 }});
```



tests > Week3Assignment > TS BasicLocatorsEditIndividuals.spec.ts > test("SalesForce Application-Edit Individuals") callback

```
import test, { expect } from "@playwright/test";
test("SalesForce Application-Edit Individuals", async ({ page }) => {
  await page.goto("https://login.salesforce.com");
  await page.locator("//input[@id='username']").fill("dilip@testleaf.com");
  await page.locator("//input[@id='password']").fill("Leaf@123");
  await page.locator("//input[@id='Login']").click();
  await page.locator(".slds-r5").waitFor({ state: 'visible' });
  await page.locator(".slds-r5").click();
  await page.locator("//button[text()='View All']").waitFor({ state: 'visible' });
  await page.locator("//button[text()='View All']").click();
  // Select the "Individuals" app/module
  await page.locator("//p[text()='Individuals']").click();
  // Click on the list view dropdown to confirm the section is interactive
  await page.locator("//span[text()='Select a List View: Individuals']").click();
  // Search for the Individual named "Aradhita" and click
  await page.locator("//input[@id='input-199']").click();
  await page.locator("//input[@id='input-199']").fill("Aradhita");
  await page.keyboard.press('Enter');
  await page.locator("//span[text()='Aradhita']").click();
  // Wait for menu to appear and click "Edit"
  await page.locator("//div[text()='Edit']").click();
  // Click the salutation dropdown
  await page.click("//div[@class='salutation compoundTLRadius compoundTRRadius compoundBorderBottom form-element__row uiMenu']//
 // Wait for the dropdown options to be visible
  await page.waitForSelector("//li[@class='uiMenuItem uiRadioMenuItem']//a[@title='Ms.']");
 // Select "Ms." from the dropdown
  await page.click("//li[@class='uiMenuItem uiRadioMenuItem']//a[@title='Ms.']");
 //Enter First Name
  await page.locator("//input[@id='2037:0']").click();
  await page.locator("//input[@id='2037:0']").fill("Naik");
 // Click Save to update the individual
  await page.click("//span[text()='Save']");
// Capture and assert the success toast message
  const Message = await page.locator("//span[contains(@class, 'toastMessage')]").textContent();
  expect(Message).toBe('Individual "Naik Aradhita" was saved.');
// Log the message to the console
  console.log(Message);{ timeout: 60000 }});
```

Screenshots



Individual "Naik Aradhita" was saved.

Individual
Ms. Naik Aradhita

We found no potential duplicates of this Individual.

No duplicate rules are activated. Activate duplicate rules to identify potential duplicate records.

Individual History (1)

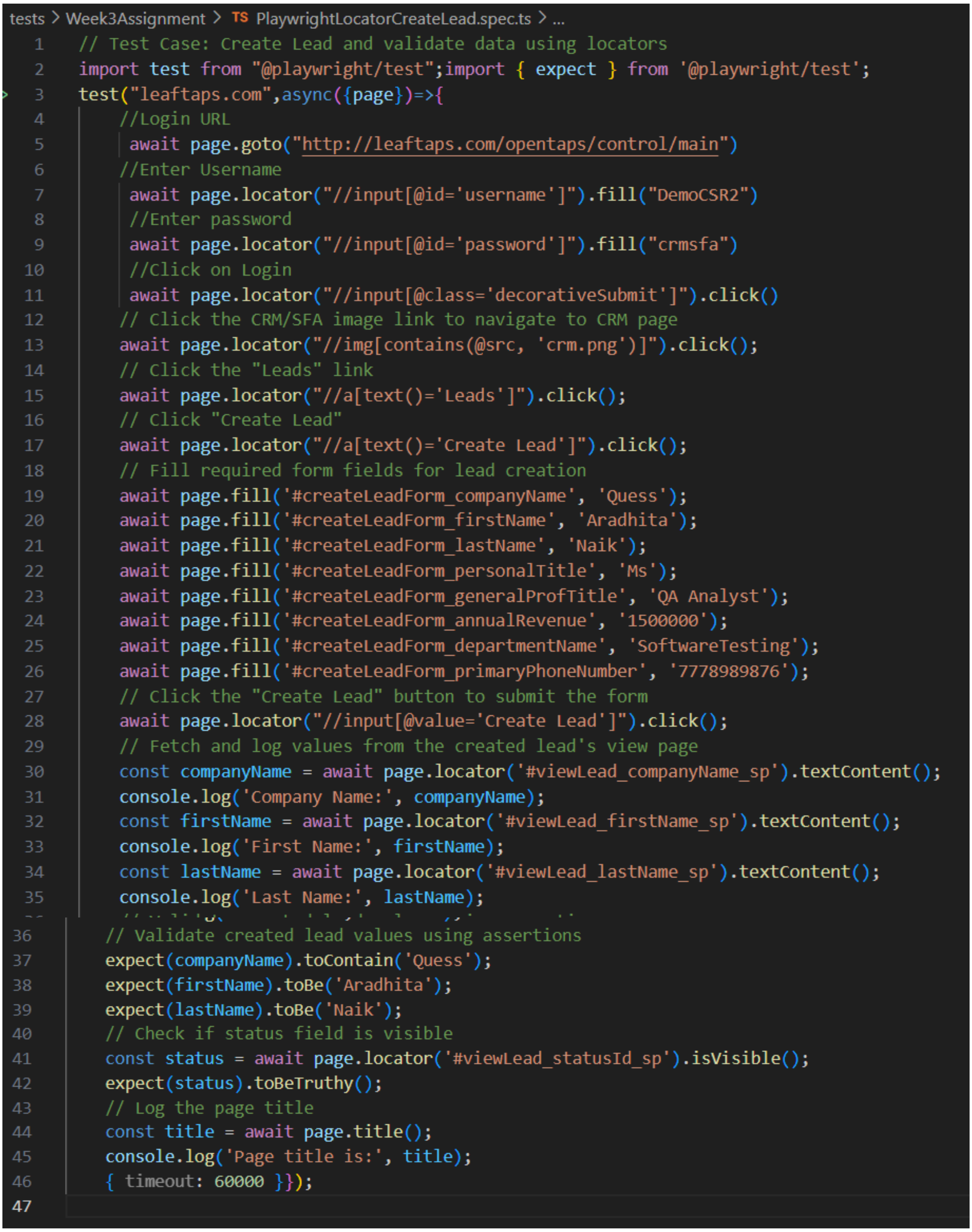
Date	Field	User	Original Value	New Value
12/07/2025, 7:55 pm	Created.	Dilip Kumar		

View All

Contacts (0)

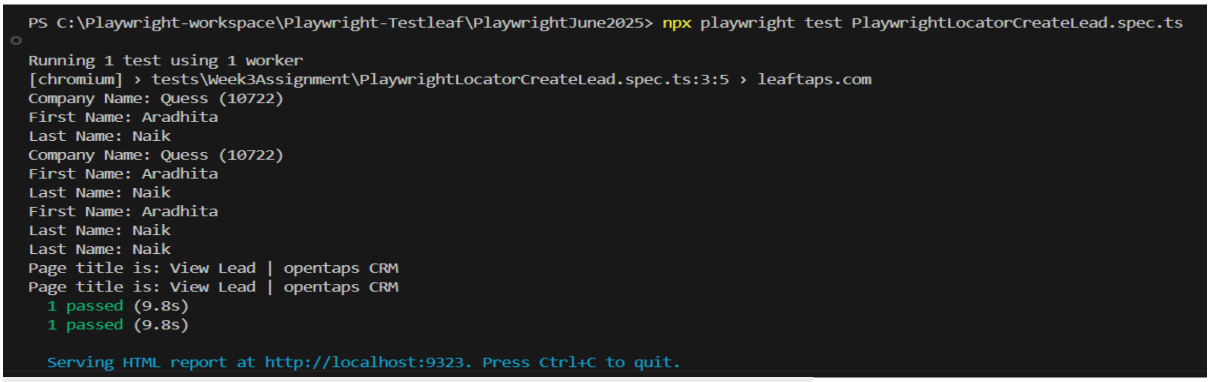
Leads (0)

5.PlaywrightLocators-CreateLead-Get the page title

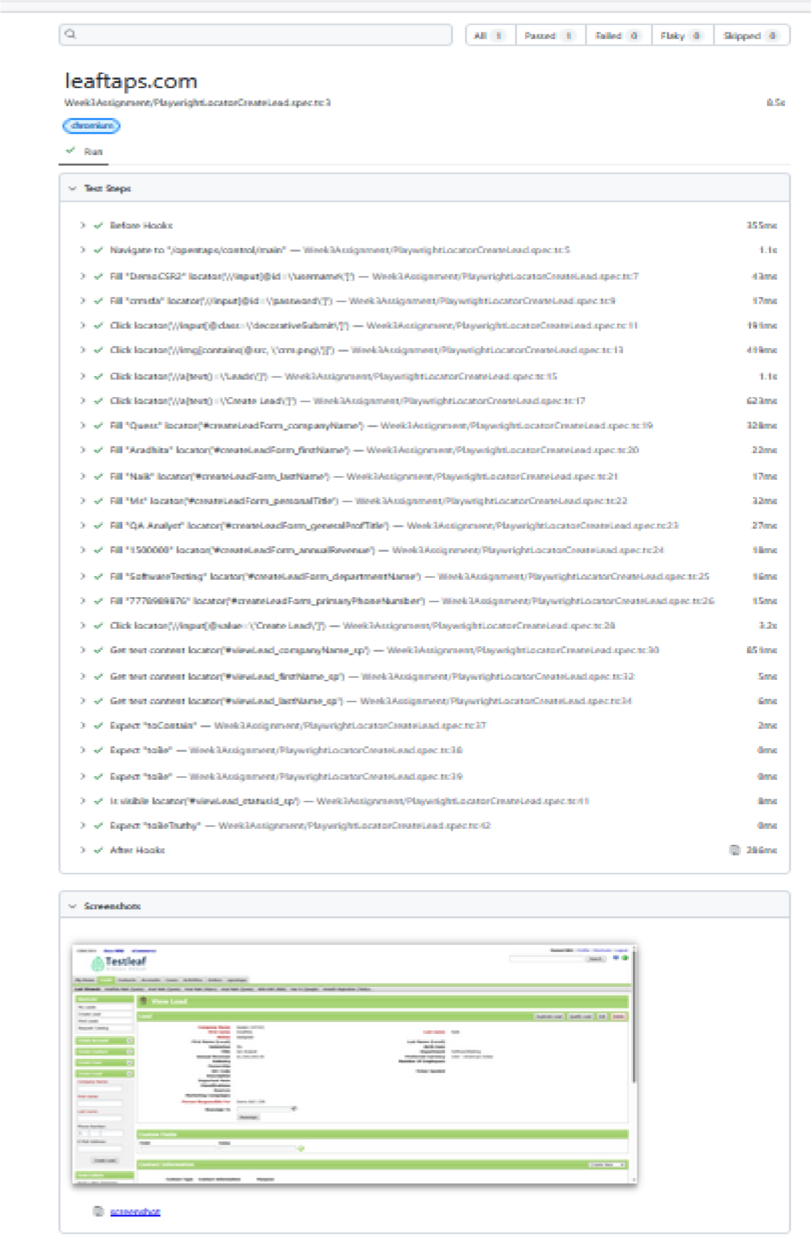


tests > Week3Assignment > TS PlaywrightLocatorCreateLead.spec.ts > ...

```
// Test Case: Create Lead and validate data using locators
import test from "@playwright/test";import { expect } from '@playwright/test';
test("leaftaps.com",async({page})=>{
    //Login URL
     await page.goto("http://leaftaps.com/opentaps/control/main")
    //Enter Username
     await page.locator("//input[@id='username']").fill("DemoCSR2")
     //Enter password
     await page.locator("//input[@id='password']").fill("crmsfa")
     //Click on Login
     await page.locator("//input[@class='decorativeSubmit']").click()
    // Click the CRM/SFA image link to navigate to CRM page
    await page.locator("//img[contains(@src, 'crm.png')]").click();
    // Click the "Leads" link
    await page.locator("//a[text()='Leads']").click();
    // Click "Create Lead"
    await page.locator("//a[text()='Create Lead']").click();
    // Fill required form fields for lead creation
    await page.fill('#createLeadForm_companyName', 'Quess');
    await page.fill('#createLeadForm_firstName', 'Aradhita');
    await page.fill('#createLeadForm_lastName', 'Naik');
    await page.fill('#createLeadForm_personalTitle', 'Ms');
    await page.fill('#createLeadForm_generalProfTitle', 'QA Analyst');
    await page.fill('#createLeadForm_annualRevenue', '1500000');
    await page.fill('#createLeadForm_departmentName', 'SoftwareTesting');
    await page.fill('#createLeadForm_primaryPhoneNumber', '7778989876');
    // Click the "Create Lead" button to submit the form
    await page.locator("//input[@value='Create Lead']").click();
    // Fetch and log values from the created lead's view page
    const companyName = await page.locator('#viewLead_companyName_sp').textContent();
    console.log('Company Name:', companyName);
    const firstName = await page.locator('#viewLead_firstName_sp').textContent();
    console.log('First Name:', firstName);
    const lastName = await page.locator('#viewLead_lastName_sp').textContent();
    console.log('Last Name:', lastName);
    // Validate created lead values using assertions
    expect(companyName).toContain('Quess');
    expect(firstName).toBe('Aradhita');
    expect(lastName).toBe('Naik');
    // Check if status field is visible
    const status = await page.locator('#viewLead_statusId_sp').isVisible();
    expect(status).toBeTruthy();
    // Log the page title
    const title = await page.title();
    console.log('Page title is:', title);
    { timeout: 60000 }});
```



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

Running 1 test using 1 worker
[chromium] › tests\Week3Assignment\PlaywrightLocatorCreateLead.spec.ts:3:5 › leaftaps.com
Company Name: Quess (10722)
First Name: Aradhita
Last Name: Naik
Company Name: Quess (10722)
First Name: Aradhita
Last Name: Naik
First Name: Aradhita
Last Name: Naik
Last Name: Naik
Page title is: View Lead | opentaps CRM
Page title is: View Lead | opentaps CRM
  1 passed (9.8s)
  1 passed (9.8s)

  Serving HTML report at http://localhost:9323. Press Ctrl+C to quit.
```

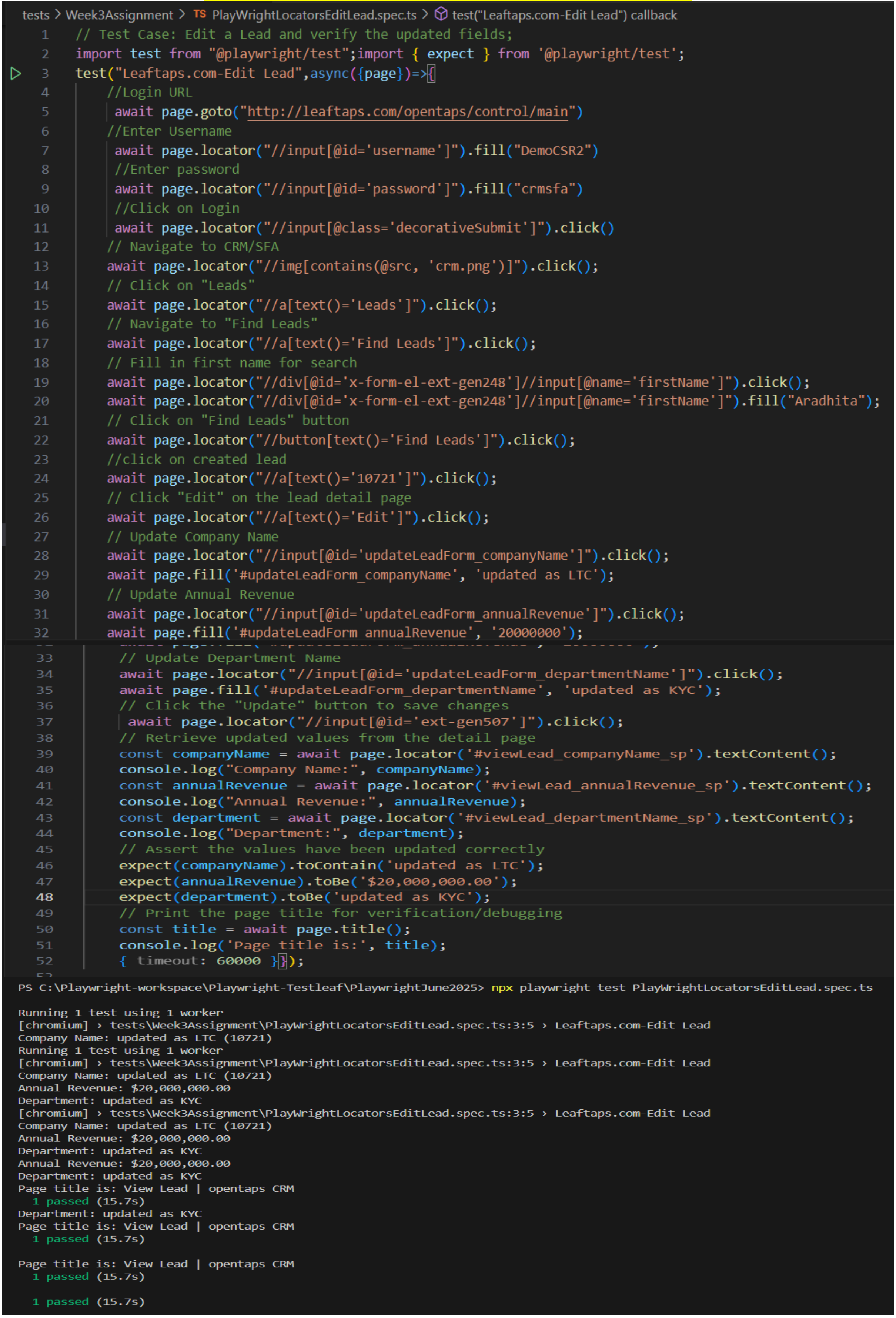


leaftaps.com

Week3Assignment/PlaywrightLocatorCreateLead.spec.ts:3 8.5s

```
Before Hooks                                                                                               355ms
Navigate to "/opentaps/control/main" — Week3Assignment/PlaywrightLocatorCreateLead.spec.ts:5                1.1s
Fill "DemoCSR2" locator('//input[@id = \'username\']') — Week3Assignment/PlaywrightLocatorCreateLead.spec.ts:7   43ms
Fill "crmsfa" locator('//input[@id = \'password\']') — Week3Assignment/PlaywrightLocatorCreateLead.spec.ts:9     17ms
Click locator('//input[@class = \'decorativeSubmit\']') — Week3Assignment/PlaywrightLocatorCreateLead.spec.ts:11  191ms
Click locator('//img[contains(@src, \'crm.png\')]') — Week3Assignment/PlaywrightLocatorCreateLead.spec.ts:13     419ms
Click locator('//a[text() = \'Leads\']') — Week3Assignment/PlaywrightLocatorCreateLead.spec.ts:15              1.1s
Click locator('//a[text() = \'Create Lead\']') — Week3Assignment/PlaywrightLocatorCreateLead.spec.ts:17        623ms
Fill "Quess" locator('#createLeadForm_companyName') — Week3Assignment/PlaywrightLocatorCreateLead.spec.ts:19    328ms
Fill "Aradhita" locator('#createLeadForm_firstName') — Week3Assignment/PlaywrightLocatorCreateLead.spec.ts:20    22ms
Fill "Naik" locator('#createLeadForm_lastName') — Week3Assignment/PlaywrightLocatorCreateLead.spec.ts:21         17ms
Fill "Mr" locator('#createLeadForm_personalTitle') — Week3Assignment/PlaywrightLocatorCreateLead.spec.ts:22      32ms
Fill "QA Analyst" locator('#createLeadForm_generalProfTitle') — Week3Assignment/PlaywrightLocatorCreateLead.spec.ts:23   27ms
Fill "1500000" locator('#createLeadForm_annualRevenue') — Week3Assignment/PlaywrightLocatorCreateLead.spec.ts:24   18ms
Fill "SoftwareTesting" locator('#createLeadForm_departmentName') — Week3Assignment/PlaywrightLocatorCreateLead.spec.ts:25   16ms
Fill "7778989876" locator('#createLeadForm_primaryPhoneNumber') — Week3Assignment/PlaywrightLocatorCreateLead.spec.ts:26   15ms
Click locator('//input[@value = \'Create Lead\']') — Week3Assignment/PlaywrightLocatorCreateLead.spec.ts:28     3.2s
Get text content locator('#viewLead_companyName_sp') — Week3Assignment/PlaywrightLocatorCreateLead.spec.ts:30   651ms
Get text content locator('#viewLead_firstName_sp') — Week3Assignment/PlaywrightLocatorCreateLead.spec.ts:32      5ms
Get text content locator('#viewLead_lastName_sp') — Week3Assignment/PlaywrightLocatorCreateLead.spec.ts:34       6ms
Expect "toContain" — Week3Assignment/PlaywrightLocatorCreateLead.spec.ts:37                                     2ms
Expect "toBe" — Week3Assignment/PlaywrightLocatorCreateLead.spec.ts:38                                          0ms
Expect "toBe" — Week3Assignment/PlaywrightLocatorCreateLead.spec.ts:39                                          0ms
Is visible locator('#viewLead_statusId_sp') — Week3Assignment/PlaywrightLocatorCreateLead.spec.ts:41              8ms
Expect "toBeTruthy" — Week3Assignment/PlaywrightLocatorCreateLead.spec.ts:42                                    0ms
After Hooks                                                                                                286ms
```

6.PlaywrightLocators-EditLead: Verify the edited fields


tests > Week3Assignment > TS PlayWrightLocatorsEditLead.spec.ts > test("Leaftaps.com-Edit Lead") callback

```
// Test Case: Edit a Lead and verify the updated fields;
import test from "@playwright/test";import { expect } from '@playwright/test';
test("Leaftaps.com-Edit Lead",async({page})=>{
    //Login URL
     await page.goto("http://leaftaps.com/opentaps/control/main")
    //Enter Username
     await page.locator("//input[@id='username']").fill("DemoCSR2")
     //Enter password
     await page.locator("//input[@id='password']").fill("crmsfa")
     //Click on Login
     await page.locator("//input[@class='decorativeSubmit']").click()
    // Navigate to CRM/SFA
    await page.locator("//img[contains(@src, 'crm.png')]").click();
    // Click on "Leads"
    await page.locator("//a[text()='Leads']").click();
    // Navigate to "Find Leads"
    await page.locator("//a[text()='Find Leads']").click();
    // Fill in first name for search
    await page.locator("//div[@id='x-form-el-ext-gen248']//input[@name='firstName']").click();
    await page.locator("//div[@id='x-form-el-ext-gen248']//input[@name='firstName']").fill("Aradhita");
    // Click on "Find Leads" button
    await page.locator("//button[text()='Find Leads']").click();
    //click on created lead
    await page.locator("//a[text()='10721']").click();
    // Click "Edit" on the lead detail page
    await page.locator("//a[text()='Edit']").click();
    // Update Company Name
    await page.locator("//input[@id='updateLeadForm_companyName']").click();
    await page.fill('#updateLeadForm_companyName', 'updated as LTC');
    // Update Annual Revenue
    await page.locator("//input[@id='updateLeadForm_annualRevenue']").click();
    await page.fill('#updateLeadForm_annualRevenue', '20000000');
    // Update Department Name
    await page.locator("//input[@id='updateLeadForm_departmentName']").click();
    await page.fill('#updateLeadForm_departmentName', 'updated as KYC');
    // Click the "Update" button to save changes
     await page.locator("//input[@id='ext-gen507']").click();
    // Retrieve updated values from the detail page
    const companyName = await page.locator('#viewLead_companyName_sp').textContent();
    console.log("Company Name:", companyName);
    const annualRevenue = await page.locator('#viewLead_annualRevenue_sp').textContent();
    console.log("Annual Revenue:", annualRevenue);
    const department = await page.locator('#viewLead_departmentName_sp').textContent();
    console.log("Department:", department);
    // Assert the values have been updated correctly
    expect(companyName).toContain('updated as LTC');
    expect(annualRevenue).toBe('$20,000,000.00');
    expect(department).toBe('updated as KYC');
    // Print the page title for verification/debugging
    const title = await page.title();
    console.log('Page title is:', title);
    { timeout: 60000 }});
```

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

Running 1 test using 1 worker
[chromium] › tests\Week3Assignment\PlayWrightLocatorsEditLead.spec.ts:3:5 › Leaftaps.com-Edit Lead
Company Name: updated as LTC (10721)
Running 1 test using 1 worker
[chromium] › tests\Week3Assignment\PlayWrightLocatorsEditLead.spec.ts:3:5 › Leaftaps.com-Edit Lead
Company Name: updated as LTC (10721)
Annual Revenue: $20,000,000.00
Department: updated as KYC
[chromium] › tests\Week3Assignment\PlayWrightLocatorsEditLead.spec.ts:3:5 › Leaftaps.com-Edit Lead
Company Name: updated as LTC (10721)
Annual Revenue: $20,000,000.00
Department: updated as KYC
Annual Revenue: $20,000,000.00
Department: updated as KYC
Page title is: View Lead | opentaps CRM
  1 passed (15.7s)
Department: updated as KYC
Page title is: View Lead | opentaps CRM
  1 passed (15.7s)

Page title is: View Lead | opentaps CRM
  1 passed (15.7s)

  1 passed (15.7s)
```

All 1 | Passed 1 | Failed 0 | Flaky 0 | Skipped 0

Leaftaps.com-Edit Lead

Week3Assignment/PlayWrightLocatorsEditLead.spec.ts:3 14.4s

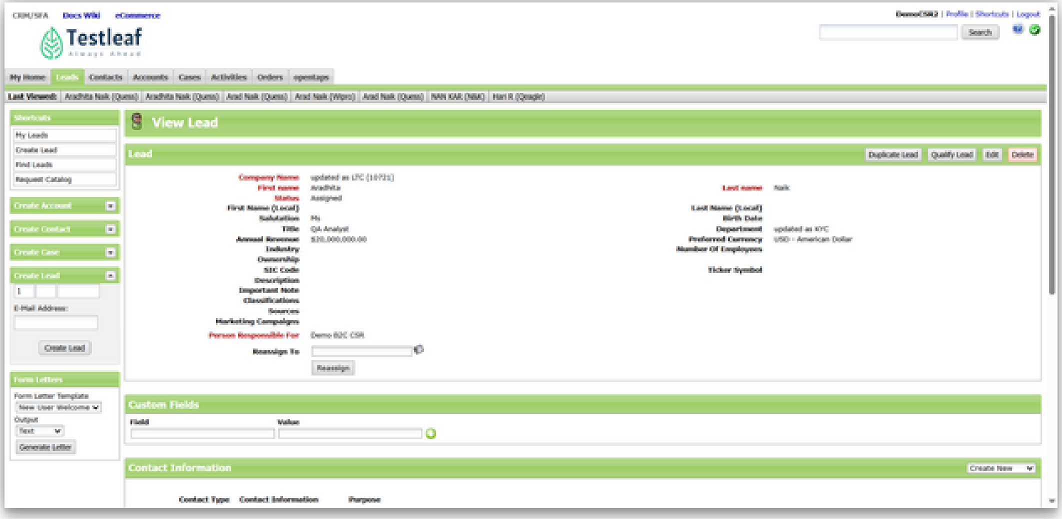
chromium

✓ Run

Test Steps

```
> ✓ Before Hooks                                                                                                         347ms
> ✓ Navigate to "/opentaps/control/main" — Week3Assignment/PlayWrightLocatorsEditLead.spec.ts:5                          1.0s
> ✓ Fill "DemoCSR2" locator('//input[@id=\'username\']') — Week3Assignment/PlayWrightLocatorsEditLead.spec.ts:7          41ms
> ✓ Fill "crmsfa" locator('//input[@id=\'password\']') — Week3Assignment/PlayWrightLocatorsEditLead.spec.ts:9            23ms
> ✓ Click locator('//input[@class=\'decorativeSubmit\']') — Week3Assignment/PlayWrightLocatorsEditLead.spec.ts:11        172ms
> ✓ Click locator('//img[contains(@src, \'crm.png\')]') — Week3Assignment/PlayWrightLocatorsEditLead.spec.ts:13          557ms
> ✓ Click locator('//a[text()=\'Leads\']') — Week3Assignment/PlayWrightLocatorsEditLead.spec.ts:15                       1.1s
> ✓ Click locator('//a[text()=\'Find Leads\']') — Week3Assignment/PlayWrightLocatorsEditLead.spec.ts:17                  566ms
> ✓ Click locator('//div[@id=\'x-form-el-ext-gen248\']//input[@name=\'firstName\']') — Week3Assignment/PlayWrightLocatorsEditLead.spe...346ms
> ✓ Fill "Aradhita" locator('//div[@id=\'x-form-el-ext-gen248\']//input[@name=\'firstName\']') — Week3Assignment/PlayWrightLocatorsEditL...25ms
> ✓ Click locator('//button[text()=\'Find Leads\']') — Week3Assignment/PlayWrightLocatorsEditLead.spec.ts:22             58ms
> ✓ Click locator('//a[text()=\'10721\']') — Week3Assignment/PlayWrightLocatorsEditLead.spec.ts:24                       396ms
> ✓ Click locator('//a[text()=\'Edit\']') — Week3Assignment/PlayWrightLocatorsEditLead.spec.ts:27                        567ms
> ✓ Click locator('//input[@id=\'updateLeadForm_companyName\']') — Week3Assignment/PlayWrightLocatorsEditLead.spec.ts:30 488ms
> ✓ Fill "updated as LTC" locator('#updateLeadForm_companyName') — Week3Assignment/PlayWrightLocatorsEditLead.spec.ts:31  21ms
> ✓ Click locator('//input[@id=\'updateLeadForm_annualRevenue\']') — Week3Assignment/PlayWrightLocatorsEditLead.spec.ts:33 65ms
> ✓ Fill "20000000" locator('#updateLeadForm_annualRevenue') — Week3Assignment/PlayWrightLocatorsEditLead.spec.ts:34     17ms
> ✓ Click locator('//input[@id=\'updateLeadForm_departmentName\']') — Week3Assignment/PlayWrightLocatorsEditLead.spec.ts:36 67ms
> ✓ Fill "updated as KYC" locator('#updateLeadForm_departmentName') — Week3Assignment/PlayWrightLocatorsEditLead.spec.ts:37 17ms
> ✓ Click locator('//input[@id=\'ext-gen507\']') — Week3Assignment/PlayWrightLocatorsEditLead.spec.ts:40                 6.0s
> ✓ Get text content locator('#viewLead_companyName_sp') — Week3Assignment/PlayWrightLocatorsEditLead.spec.ts:43         2.2s
> ✓ Get text content locator('#viewLead_annualRevenue_sp') — Week3Assignment/PlayWrightLocatorsEditLead.spec.ts:45       17ms
> ✓ Get text content locator('#viewLead_departmentName_sp') — Week3Assignment/PlayWrightLocatorsEditLead.spec.ts:47      26ms
> ✓ Expect "toContain" — Week3Assignment/PlayWrightLocatorsEditLead.spec.ts:51                                           2ms
> ✓ Expect "toBe" — Week3Assignment/PlayWrightLocatorsEditLead.spec.ts:52                                                1ms
> ✓ Expect "toBe" — Week3Assignment/PlayWrightLocatorsEditLead.spec.ts:53                                                1ms
> ✓ After Hooks                                                                                                          296ms
```

Screenshots



View Lead

Lead

Company Name	updated as LTC (10721)		
First name	Aradhita	Last name	Naik
Status	Assigned		
First Name (Local)		Last Name (Local)	
Salutation	Ms	Birth Date	
Title	QA Analyst	Department	updated as KYC
Annual Revenue	$20,000,000.00	Preferred Currency	USD - American Dollar
Industry		Number Of Employees	
Ownership			
SIC Code		Ticker Symbol	
Description			
Important Note			
Classifications			
Sources			
Marketing Campaigns			
Person Responsible For	Demo B2C CSR		
Reassign To			

Custom Fields

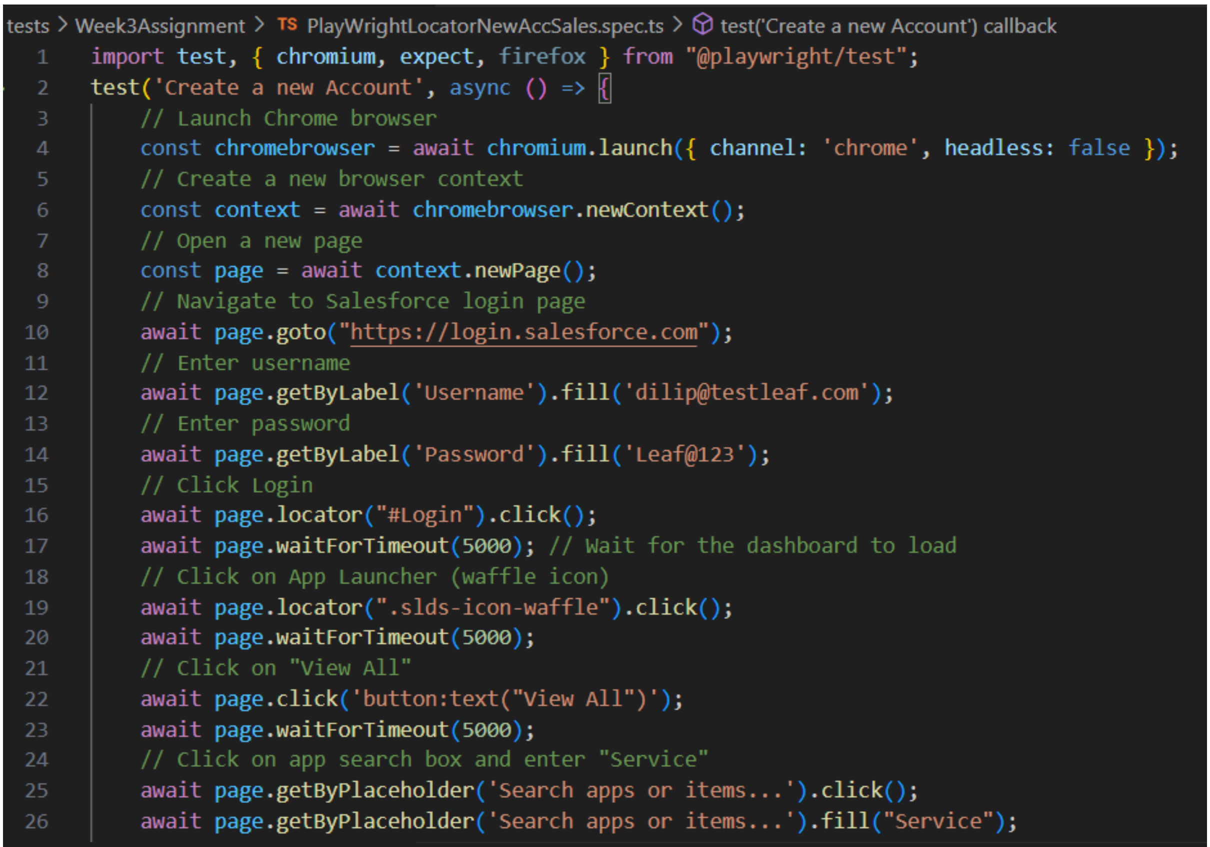
Field	Value

Contact Information

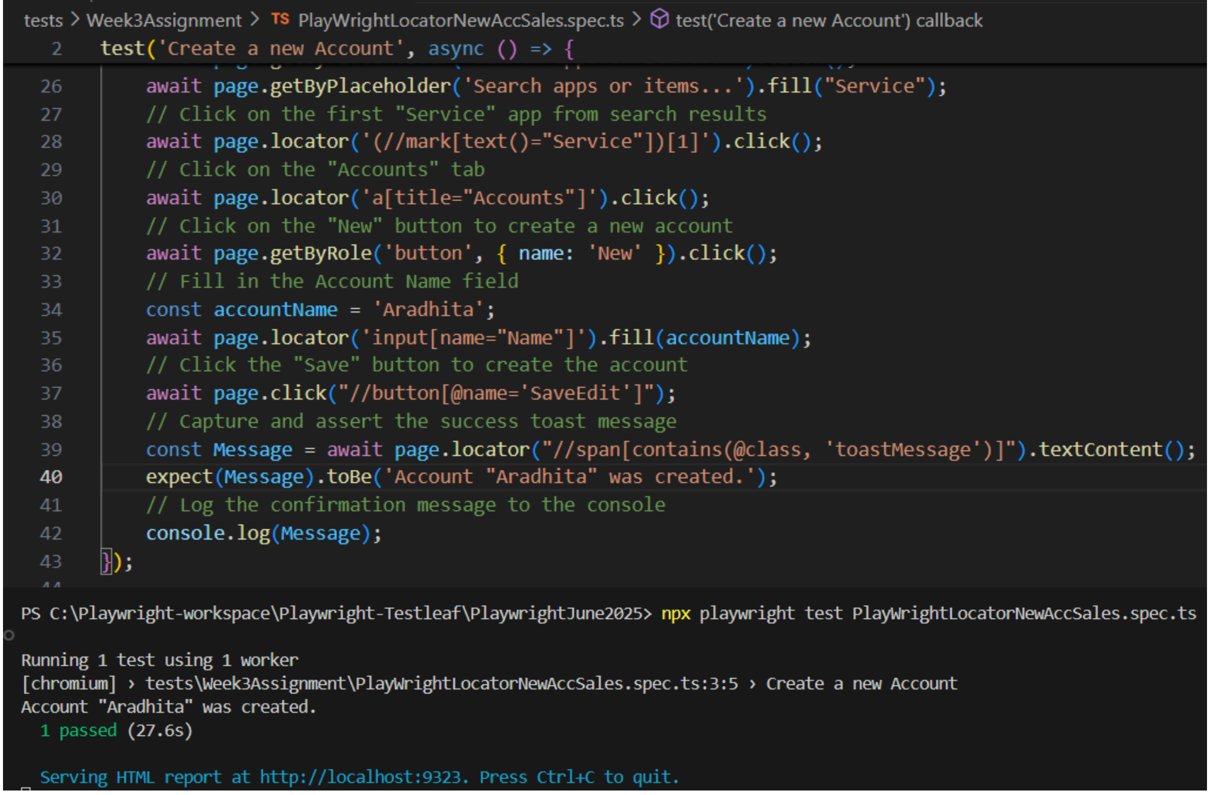
Contact Type	Contact Information	Purpose

screenshot

7. Create a new Account-Verify the toast message displayed

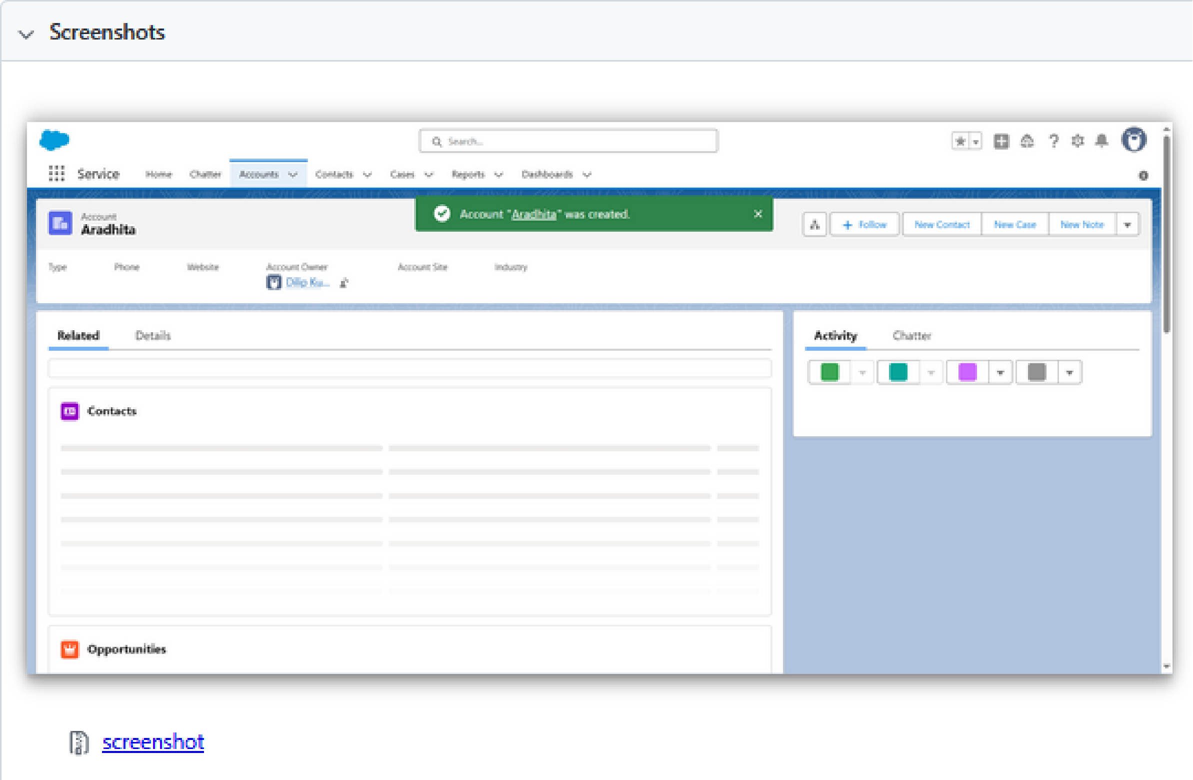

tests > Week3Assignment > TS PlayWrightLocatorNewAccSales.spec.ts > test('Create a new Account') callback

```
import test, { chromium, expect, firefox } from "@playwright/test";
test('Create a new Account', async () => {
    // Launch Chrome browser
    const chromebrowser = await chromium.launch({ channel: 'chrome', headless: false });
    // Create a new browser context
    const context = await chromebrowser.newContext();
    // Open a new page
    const page = await context.newPage();
    // Navigate to Salesforce login page
    await page.goto("https://login.salesforce.com");
    // Enter username
    await page.getByLabel('Username').fill('dilip@testleaf.com');
    // Enter password
    await page.getByLabel('Password').fill('Leaf@123');
    // Click Login
    await page.locator("#Login").click();
    await page.waitForTimeout(5000); // Wait for the dashboard to load
    // Click on App Launcher (waffle icon)
    await page.locator(".slds-icon-waffle").click();
    await page.waitForTimeout(5000);
    // Click on "View All"
    await page.click('button:text("View All")');
    await page.waitForTimeout(5000);
    // Click on app search box and enter "Service"
    await page.getByPlaceholder('Search apps or items...').click();
    await page.getByPlaceholder('Search apps or items...').fill("Service");
```


tests > Week3Assignment > TS PlayWrightLocatorNewAccSales.spec.ts > test('Create a new Account') callback

```
test('Create a new Account', async () => {
    await page.getByPlaceholder('Search apps or items...').fill("Service");
    // Click on the first "Service" app from search results
    await page.locator('(//mark[text()="Service"])[1]').click();
    // Click on the "Accounts" tab
    await page.locator('a[title="Accounts"]').click();
    // Click on the "New" button to create a new account
    await page.getByRole('button', { name: 'New' }).click();
    // Fill in the Account Name field
    const accountName = 'Aradhita';
    await page.locator('input[name="Name"]').fill(accountName);
    // Click the "Save" button to create the account
    await page.click("//button[@name='SaveEdit']");
    // Capture and assert the success toast message
    const Message = await page.locator("//span[contains(@class, 'toastMessage')]").textContent();
    expect(Message).toBe('Account "Aradhita" was created.');
    // Log the confirmation message to the console
    console.log(Message);
});
```

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

Running 1 test using 1 worker
[chromium] › tests\Week3Assignment\PlayWrightLocatorNewAccSales.spec.ts:3:5 › Create a new Account
Account "Aradhita" was created.
  1 passed (27.6s)

  Serving HTML report at http://localhost:9323. Press Ctrl+C to quit.
```



Screenshots

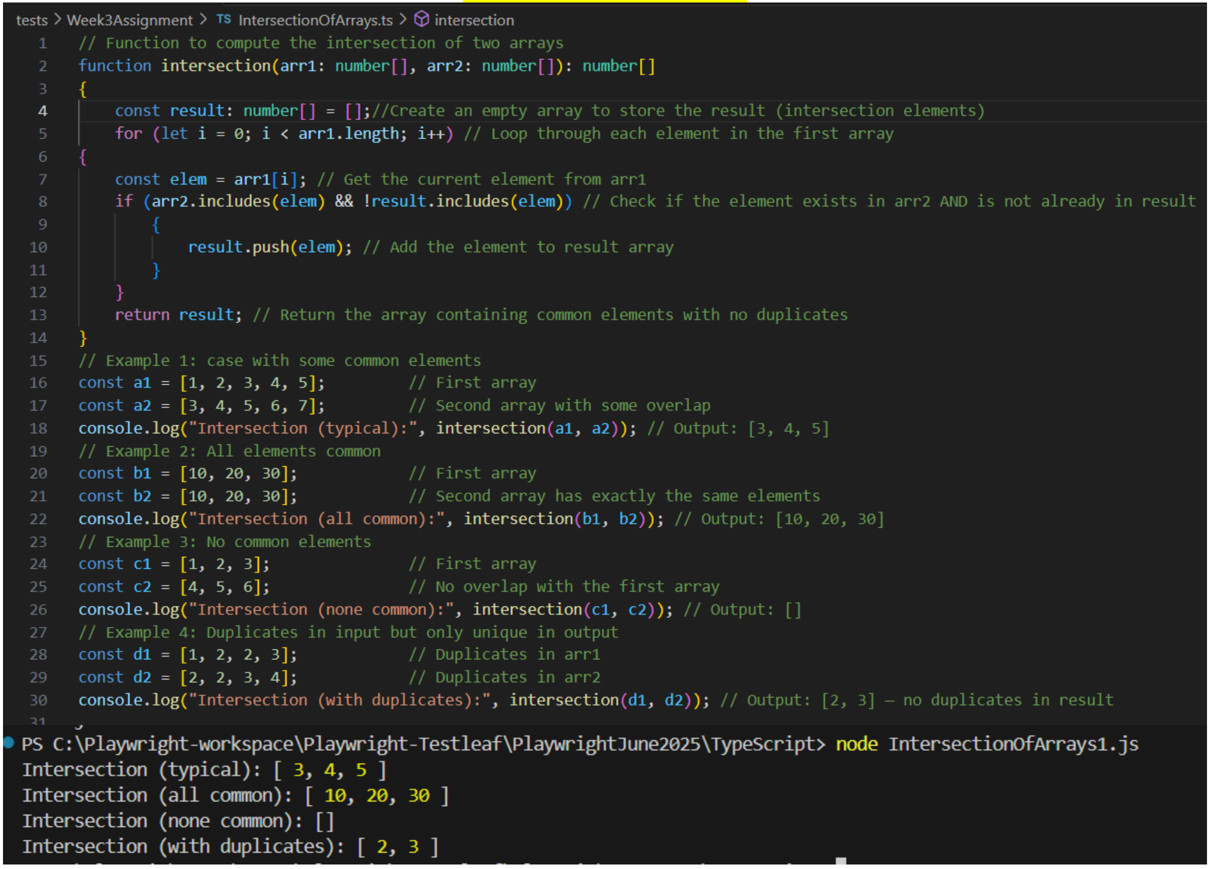
Account "Aradhita" was created.

Account
Aradhita

Account Owner: Dilip Ku...

screenshot

8.Intersection of Arrays



tests > Week3Assignment > TS IntersectionOfArrays.ts > intersection

```
// Function to compute the intersection of two arrays
function intersection(arr1: number[], arr2: number[]): number[]
{
    const result: number[] = [];//Create an empty array to store the result (intersection elements)
    for (let i = 0; i < arr1.length; i++) // Loop through each element in the first array
{
    const elem = arr1[i]; // Get the current element from arr1
    if (arr2.includes(elem) && !result.includes(elem)) // Check if the element exists in arr2 AND is not already in result
        {
            result.push(elem); // Add the element to result array
        }
    }
    return result; // Return the array containing common elements with no duplicates
}
// Example 1: case with some common elements
const a1 = [1, 2, 3, 4, 5];         // First array
const a2 = [3, 4, 5, 6, 7];         // Second array with some overlap
console.log("Intersection (typical):", intersection(a1, a2)); // Output: [3, 4, 5]
// Example 2: All elements common
const b1 = [10, 20, 30];            // First array
const b2 = [10, 20, 30];            // Second array has exactly the same elements
console.log("Intersection (all common):", intersection(b1, b2)); // Output: [10, 20, 30]
// Example 3: No common elements
const c1 = [1, 2, 3];               // First array
const c2 = [4, 5, 6];               // No overlap with the first array
console.log("Intersection (none common):", intersection(c1, c2)); // Output: []
// Example 4: Duplicates in input but only unique in output
const d1 = [1, 2, 2, 3];            // Duplicates in arr1
const d2 = [2, 2, 3, 4];            // Duplicates in arr2
console.log("Intersection (with duplicates):", intersection(d1, d2)); // Output: [2, 3] — no duplicates in result
```

```
PS C:\Playwright-workspace\Playwright-Testleaf\PlaywrightJune2025\TypeScript> node IntersectionOfArrays1.js
Intersection (typical): [ 3, 4, 5 ]
Intersection (all common): [ 10, 20, 30 ]
Intersection (none common): []
Intersection (with duplicates): [ 2, 3 ]
```