

Lesson 03 Demo 02

Work with the Locator Functions

Objective: To work with the locator functions

Tools required: Visual Studio Code

Prerequisites: Angular

Steps to be followed:

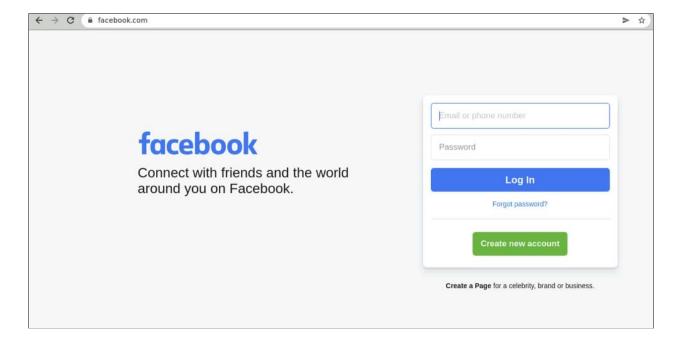
1. Inspect the Facebook login form

2. Work with the id tag

3. Get the HTML element by a class name

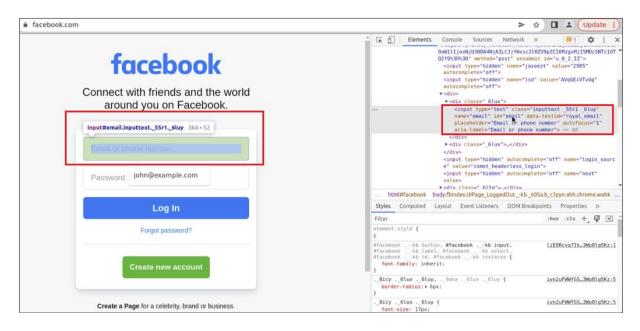
Step 1: Inspect the Facebook login form

1.1 Open **facebook.com** in the browser



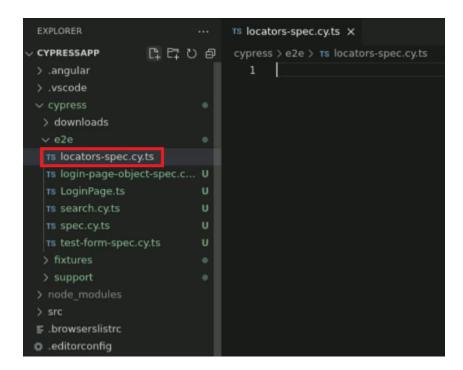


1.2 Inspect the element and check the id tag for all the fields



Step 2: Work with the id tag

2.1 Create a locators-spec.cy.ts file as shown in the given screenshot:





2.2 Copy and paste describe function from some other spec file

```
cypress > e2e > Ts locators-spec.cy.ts

describe()'empty spec', () => {
   it('passes', () => {
      cy.visit('https://google.co.in')
   })
}
```

2.3 Make changes in the file as shown in the given screenshot:

```
rs locators-spec.cy.ts U •

cypress > e2e > rs locators-spec.cy.ts > ② describe('Locators Demo Spec') callback > ② it(

describe('Locators Demo Spec', () => {
    it('types the email and password into login form', () => {
        cy.visit('https://www.facebook.com/', 4
        })
    }
}
```

2.4 Create a cy.get function and enter #email

```
rs locators-spec.cy.ts U x

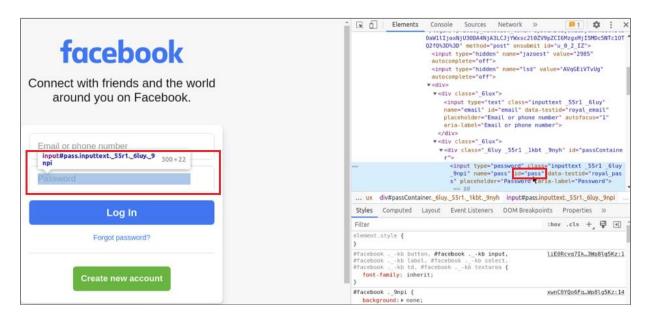
cypress > e2e > rs locators-spec.cy.ts > ② describe('Locators Demo Spec') callback > ③ it('t

describe('Locators Demo Spec', () => {
    it('types the email and password into login form', () => {
    cy.visit('https://www.facebook.com/')
    cy.get('#email').type("john@example.com')
}

cy.get('#email').type("john@example.com')
}
```



2.5 Inspect the element for the password field



2.6 Create the same function for the password field

```
rs locators-spec.cy.ts U x

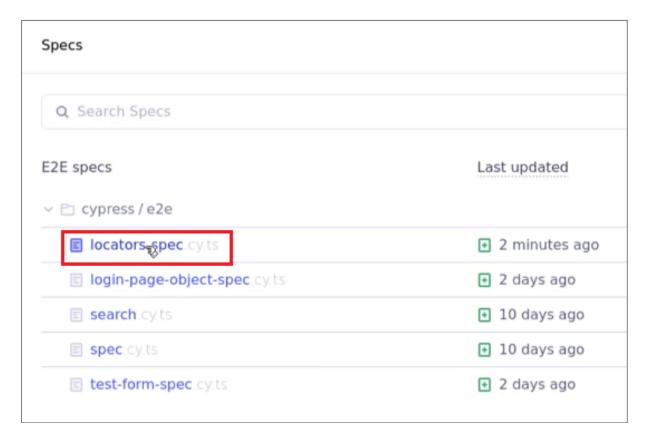
cypress > e2e > rs locators-spec.cy.ts > ② describe('Locators Demo Spec') callback > ② it('ty
    describe('Locators Demo Spec', () => {
        it('types the email and password into login form', () => {
            cy.visit('https://www.facebook.com/')
            cy.get('#email').type('john@example.com')
            cy.get('#pass').type('john123')
            }
        }
        }
    }
}
```

2.7 Run the ng e2e command

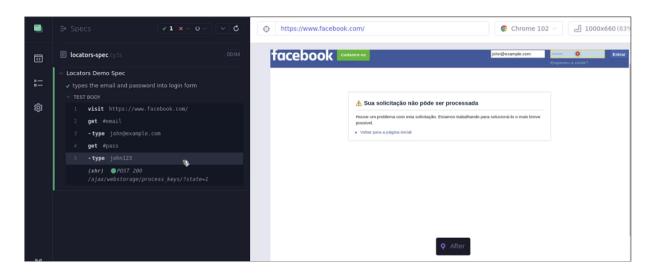
```
Node.js version v17.2.0 detected.
Odd numbered Node.js versions will not enter LTS status and should is odejs.org/en/about/releases/.
srijanighataksi@ip-172-31-27-104:-/Documents/CypressApp$ ng e2e
Node.js version v17.2.0 detected.
Odd numbered Node.js versions will not enter LTS status and should is odejs.org/en/about/releases/.
"Generating browser application bundles (phase: setup)...
```



2.8 Select the spec



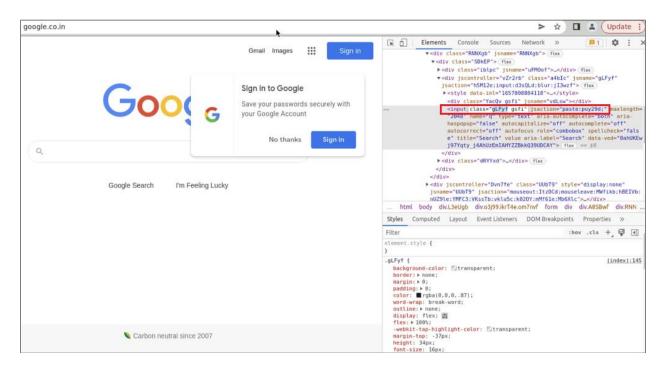
2.9 Check the test results





Step 3: Get the HTML element by the class name

3.1 Go to **google.co.in**, and then copy the class value from the inspect element of the search field



3.2 Write a test case for google.co.in and paste the class value in it

```
rs locators-spec.cy.ts U x

cypress > e2e > rs locators-spec.cy.ts > @ describe('Locators Demo Spec') callback > @ it('I

describe('Locators Demo Spec', () => {
    it('types the email and password into login form', () => {
        cy.visit('https://www.facebook.com/')
        cy.get('#email').type('john@example.com')
        cy.get('#pass').type('john123')
    })

it('locates search input field by css', ()=>{
    cy.visit('https://www.google.co.in/')
    cy.get('.gLFyf').type[['simplilearn']]
}

cy.get('.gLFyf').type[['simplilearn']]
}
```



3.3 Confirm the test on Cypress Launchpad

