

## Lesson 04 Demo 01

### Creating Page Object Model

**Objective:** To create Page Object Model in a Cypress project

**Tools required:** Visual Studio Code

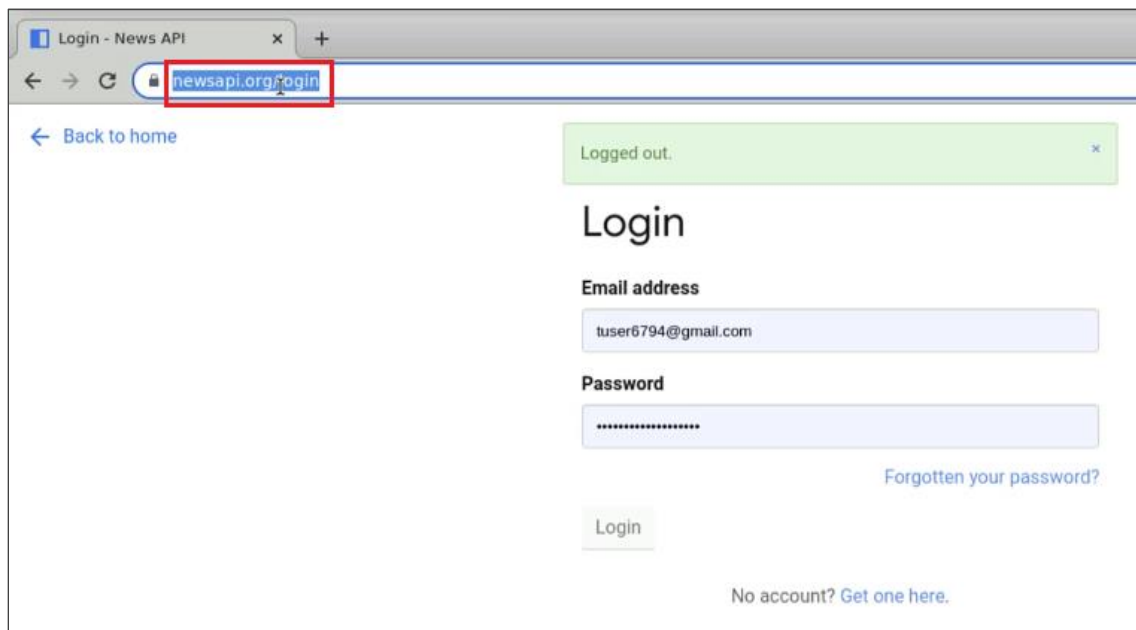
**Prerequisites:** Angular

Steps to be followed:

1. Create LoginPage.ts
2. Create login-page-object-spec.cy.ts
3. Run the test

#### Step 1: Create LoginPage.ts

##### 1.1 Copy the URL [newsapi.org/login](https://newsapi.org/login)



1.2 In the **CYPRESSAPP** project, create a **LoginPage.ts** file. Paste the URL inside the **visit**.

```

File Edit Selection View Go Run Terminal Help

EXPLORER
  CYPRESSAPP
    .angular
    .vscode
    cypress
      downloads
      e2e
        ts LoginPage.ts
        ts search.cy.ts
        ts spec.cy.ts
        ts test-form-spec.cy.ts
      fixtures
      support
      node_modules
      src
      .browserslistrc

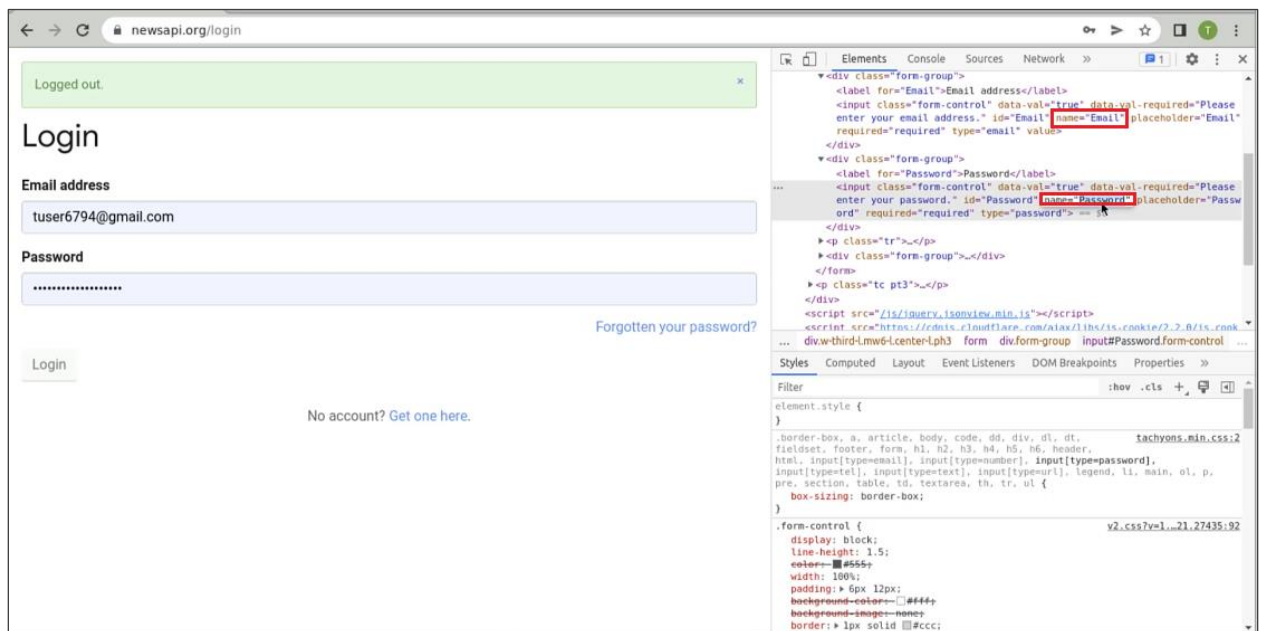
ts LoginPage.ts U
  1 class LoginPage{
  2
  3   visit(){
  4     cy.visit('https://newsapi.org/login')
  5   }
  6
  7 }
  8
  9
 10 export default LoginPage;
  
```

1.3 Similarly, create other functions

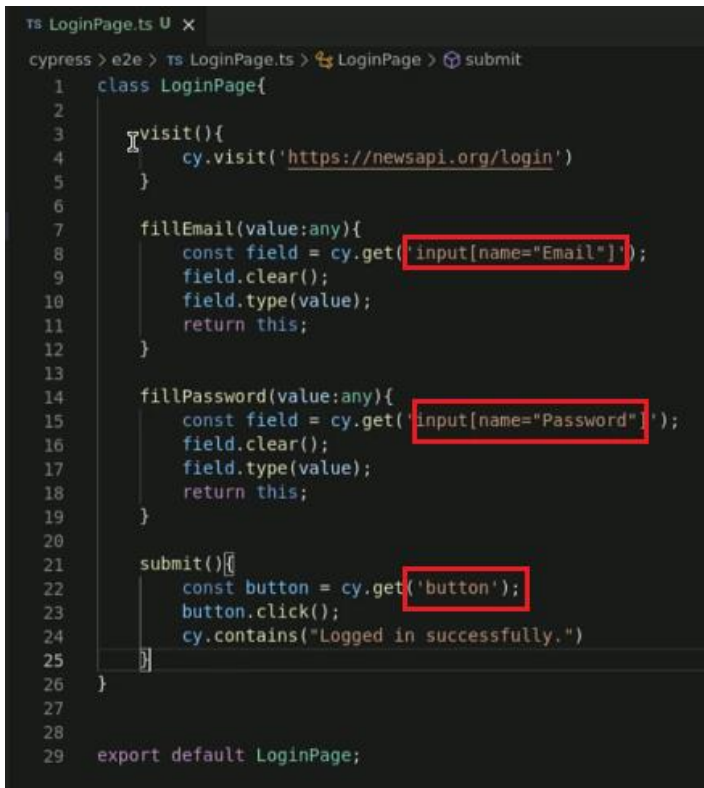
```

ts LoginPage.ts U X
  cypress > e2e > ts LoginPage.ts > LoginPage > submit
  1 class LoginPage{
  2
  3   visit(){
  4     cy.visit('https://newsapi.org/login')
  5   }
  6
  7   fillEmail(value:any){
  8     const field = cy.get('input[name="Email"]');
  9     field.clear();
 10     field.type(value);
 11     return this;
 12   }
 13
 14   fillPassword(value:any){
 15     const field = cy.get('input[name="Password"]');
 16     field.clear();
 17     field.type(value);
 18     return this;
 19   }
 20
 21   submit(){
 22     const button = cy.get('button');
 23     button.click();
 24     cy.contains("Logged in successfully.")
 25   }
 26 }
 27
 28
 29 export default LoginPage;
  
```

## 1.4 Inspect all the elements. Copy name values for **email** and **password** functions.

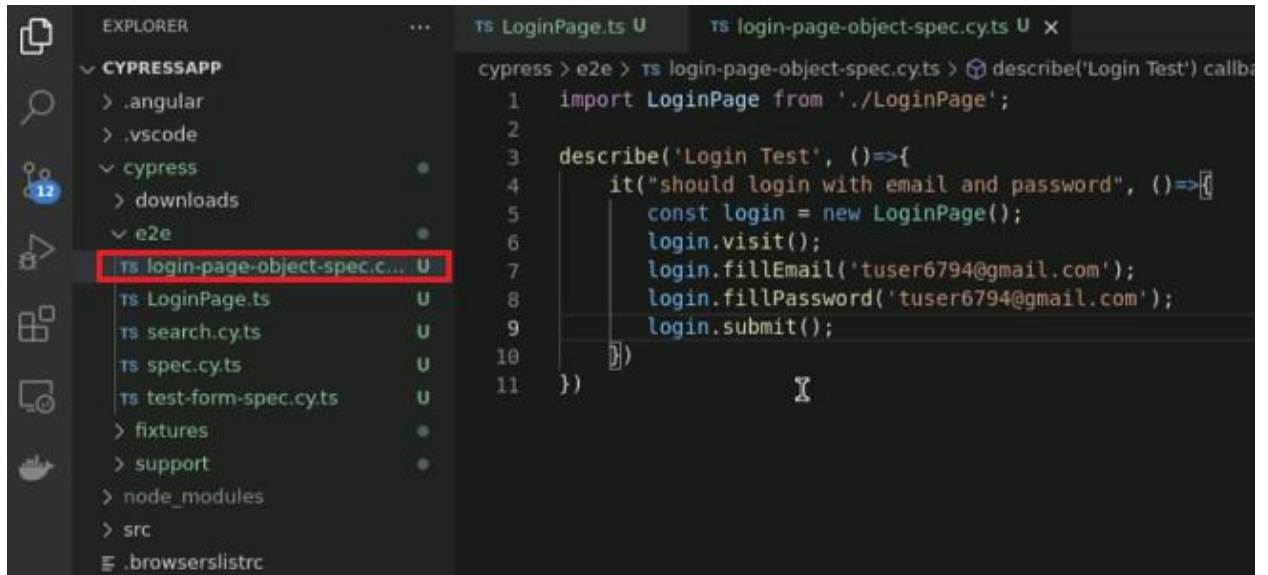


## 1.5 Paste those values in the **cy.get** functions



## Step 2: Create login-page-object-spec.cy.ts

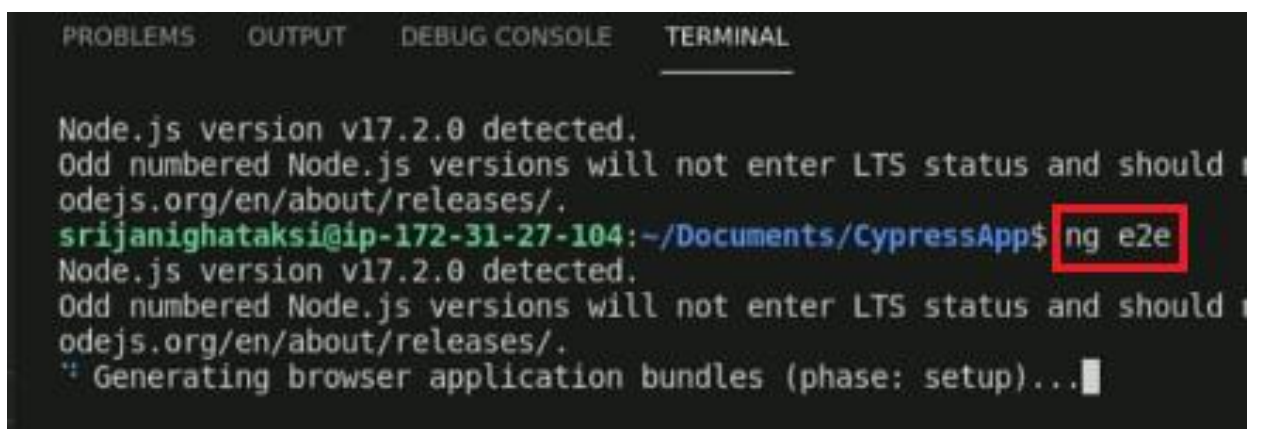
### 2.1 Create login-page-object-spec.cy.ts file



```
cyress > e2e > ts login-page-object-spec.cy.ts > describe('Login Test') callba
1  import LoginPage from './LoginPage';
2
3  describe('Login Test', ()=>{
4      it("should login with email and password", ()=>{
5          const login = new LoginPage();
6          login.visit();
7          login.fillEmail('tuser6794@gmail.com');
8          login.fillPassword('tuser6794@gmail.com');
9          login.submit();
10     })
11 })
```

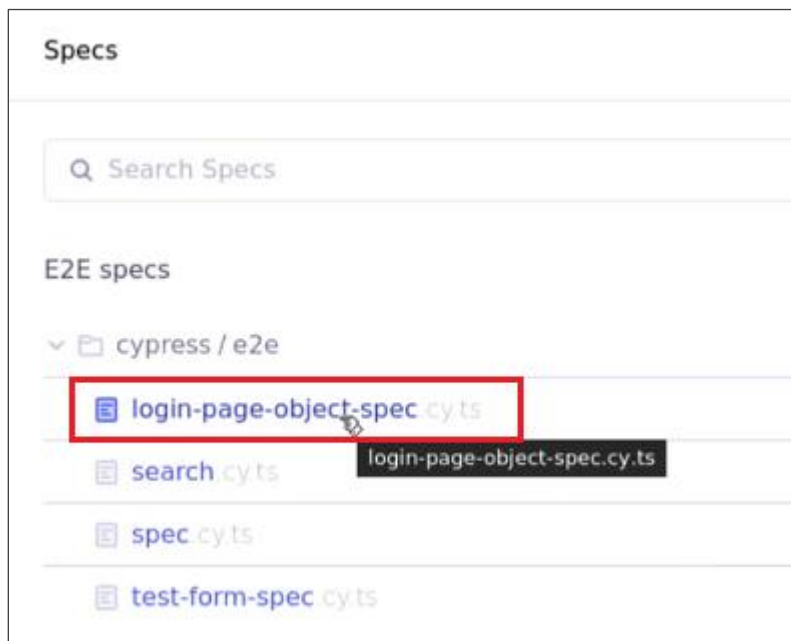
## Step 3: Run the Test

### 3.1 Run the ng e2e command

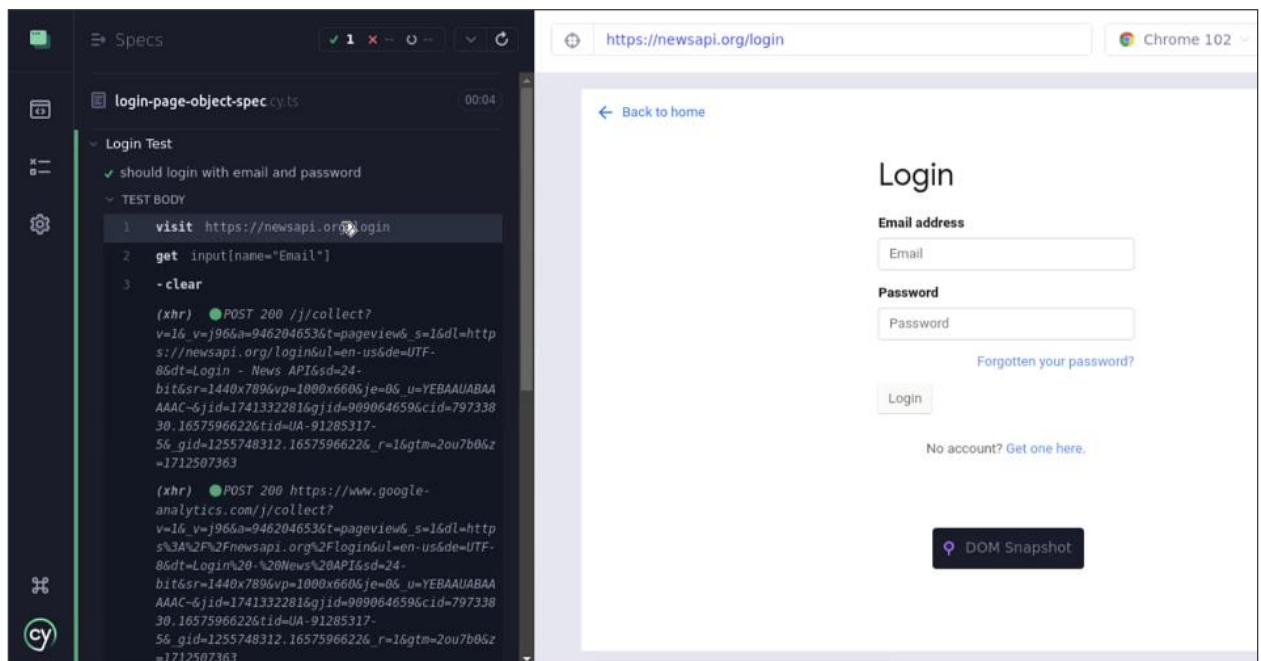


```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL
Node.js version v17.2.0 detected.
Odd numbered Node.js versions will not enter LTS status and should
odejs.org/en/about/releases/.
srijanighataksi@ip-172-31-27-104:~/Documents/CypressApp$ ng e2e
Node.js version v17.2.0 detected.
Odd numbered Node.js versions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

### 3.2 Select the spec



### 3.3 Check the test results



The image shows a Cypress test runner interface on the left and a browser window on the right. The browser window displays the 'Account' page of newsapi.org, which includes fields for Email, Password, and API key, along with buttons for 'Update email address', 'Change password', and 'DOM Snapshot'. A green notification bar at the top of the browser window says 'Logged in successfully.' The Cypress test runner on the left shows a spec file named 'login-page-object-spec.cypress.js' with a test suite 'login-page-object-spec' and a single test 'contains Logged in successfully.' The test is passing, as indicated by the green checkmark and the 'contains' command output.

```

login-page-object-spec.cypress.js
00:04

=1712507363
4   - type: tuser6794@gmail.com
5   get input[name="Password"]
6   - clear
7   - type: tuser6794@gmail.com
8   get button
9   - click
  (form sub) --submitting form--
  (page load) --page loaded--
  (new url) https://newsapi.org/account
10  contains Logged in successfully.

(fetch) GET 200
https://syndication.twitter.com/settings?
session_id=850e73aa62c32698e0f38b757323055236f
894cf

(fetch) GET 200
https://syndication.twitter.com/settings?
session_id=b5e44f485079ac9483b6c08f3b46d3a4329
e4181

(fetch) GET 200
https://syndication.twitter.com/settings?

```