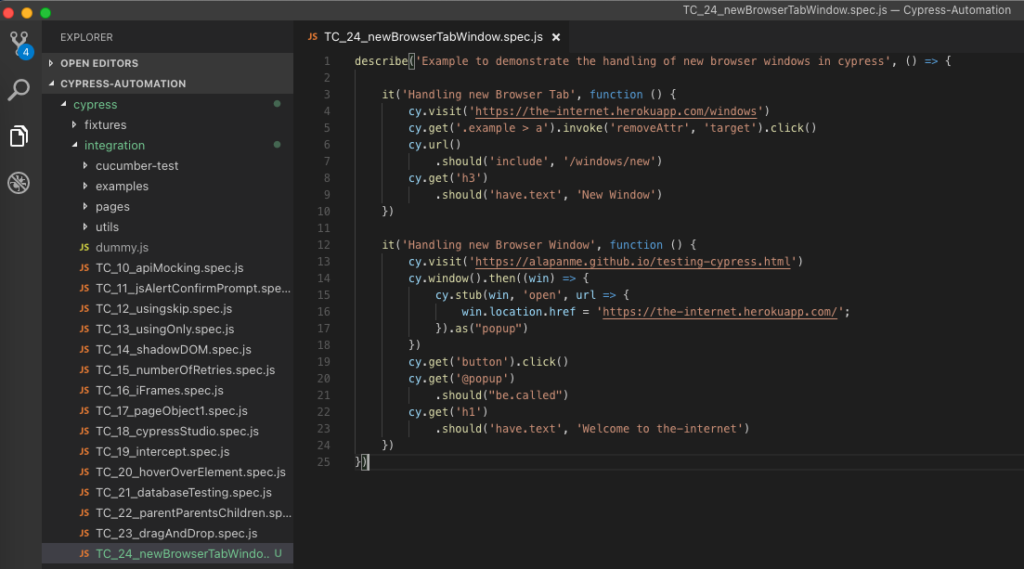
**Test Case 1:**  
Step 1: Go to <https://the-internet.herokuapp.com/windows>  
Step 2: Click on the ‘Click Here’ button. This opens a new Tab.  
Step 3: Go to the new tab and verify that the URL has the text ‘/windows/new’ and the webpage has the text ‘New Window’

**Test Case 2:**  
Step 1: Go to <https://alapanme.github.io/testing-cypress.html>  
Step 2: Click on the ‘Try it’ button.  
Step 3: A new window would be opened. Verify that the header text in the new window is ‘Welcome to the-internet’.

|  |  |
| --- | --- |
| 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 | describe('Example to demonstrate the handling of new browser windows in cypress', () => {      it('Handling new Browser Tab', function () {         cy.visit('https://the-internet.herokuapp.com/windows')         cy.get('.example > a').invoke('removeAttr', 'target').click()         cy.url()             .should('include', '/windows/new')         cy.get('h3')             .should('have.text', 'New Window')     })      it('Handling new Browser Window', function () {         cy.visit('https://alapanme.github.io/testing-cypress.html')         cy.window().then((win) => {             cy.stub(win, 'open', url => {                 win.location.href = 'https://the-internet.herokuapp.com/';             }).as("popup")         })         cy.get('button').click()         cy.get('@popup')             .should("be.called")         cy.get('h1')             .should('have.text', 'Welcome to the-internet')     }) }) |

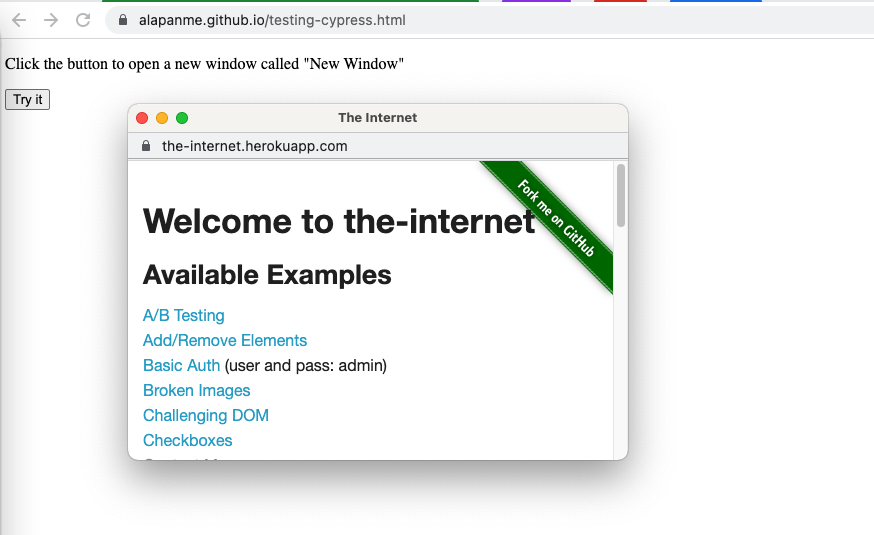
[](https://testersdock.b-cdn.net/wp-content/uploads/2021/08/cypress-new-window-and-tabs-test-script.png)

|  |  |
| --- | --- |
| 1 2 3 4 5 6 7 8 | it('Handling new Browser Tab', function() {   cy.visit('https://the-internet.herokuapp.com/windows')   cy.get('.example > a').invoke('removeAttr', 'target').click()   cy.url()     .should('include', '/windows/new')   cy.get('h3')     .should('have.text', 'New Window') }) |

Since cypress doesn’t allow the opening of a new browser tab, the workaround in such cases is to remove the attribute – **target=’\_blank’**. Now when we click the ‘Click Here’ button, instead of a new tab, the webpage is opened in the current tab. And then when the webpage is loaded, we are validating that the URL has the text ‘/windows/new’ and the webpage has the text ‘new Window’.

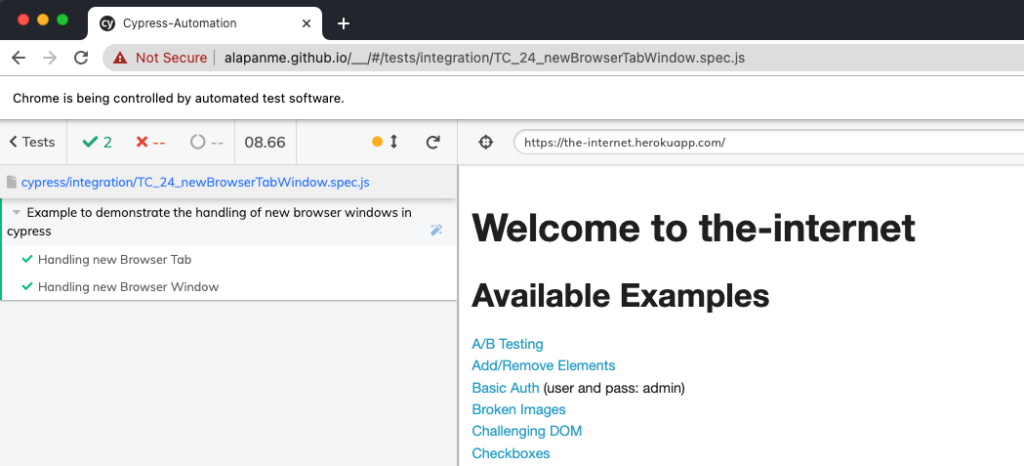
|  |  |
| --- | --- |
| 1 2 3 4 5 6 7 8 9 10 11 12 13 | it('Handling new Browser Window', function() {   cy.visit('https://alapanme.github.io/testing-cypress.html')   cy.window().then((win) => {     cy.stub(win, 'open', url => {       win.location.href = 'https://the-internet.herokuapp.com/';     }).as("popup")   })   cy.get('button').click()   cy.get('@popup')     .should("be.called")   cy.get('h1')     .should('have.text', 'Welcome to the-internet') }) |

In this test, our goal is to read the contents of the webpage that is opened after clicking the ‘Try it’ button. The webpage is opened in a new browser window.

[](https://testersdock.b-cdn.net/wp-content/uploads/2021/08/new-browser-window.png)

First, we get the Window object and then we replace the **window.open(url, target)** function with our own arrow function. Then we change the window location **win.location.href** to be the same as the popup URL and then we create an alias for the same as **popup**, so that we can refer to it later using **@popup**. Then we click the ‘Try it’ button, which triggers javascript’s **window.open()** call. To make sure that the window.open function call is triggered we are writing an assertion as **cy.get(‘@popup’).should(“be.called”)**. Then we are continuing the testing for the new “popup tab” inside the same tab and asserting that the webpage has the text ‘Welcome to the-internet’.

After Execution, We should see a ‘PASS’.

[](https://testersdock.b-cdn.net/wp-content/uploads/2021/08/cypress-new-browser-window-tab-test-script-execution.png)

Do check out 🙂

However, there are many ways this functionality can be tested in Cypress now. You can split up your tests into separate pieces and still have confidence that your application is covered.

1. Write a test to check that when performing the action in your app, the window.open event is called by using [cy.spy()](https://on.cypress.io/spy) to listen for a window.open event.

cy.visit('http://localhost:3000', {

onBeforeLoad(win) {

cy.stub(win, 'open')

}

})

// Do the action in your app like cy.get('.open-window-btn').click()

cy.window().its('open').should('be.called')

1. In a new test, use [cy.visit()](https://on.cypress.io/visit) to go to the url that would have opened in the new window, fill in the fields and click the buttons like you would in a Cypress test.

cy.visit('http://localhost:3000/new-window')

// Do the actions you want to test in the new window

<https://github.com/cypress-io/cypress-example-recipes/tree/94ae974f41ef579cdd4cb184ed852b623ef252f9/examples/stubbing-spying__window>