**What is Cypress?**

· Cypress is a next generation front end Automation testing tool built for the modern web applications

**How Cypress is Unique from other Automation tools?**

Cypress automatically waits for commands and assertions before moving on. No more async hell.

Ability to test Edge Testcases by Mocking the server response

Cypress takes snapshots as your tests run. We can hover over each commands in the Command Log to see exactly what happened at each step.

Because of its Architectural design, Cypress delivers fast, consistent and reliable test execution compared to other Automation tools

View videos of your entire tests execution when run from the Cypress Dashboard.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Cypress built on Node.js and comes packaged as an npm module,

As it is built on Node.js, It uses JavaScript for writing tests. But 90% of coding can be done using Cypress inbuilt commands which are easy to understand.

Cypress also bundles with jQuery and inherits many of jQuery methods for UI components Identification

**Cypress Architecture**

Most testing tools (like Selenium) operate by running outside of the browser and executing remote commands across the network. But Cypress engine directly operates inside the browser. In other words, It is the browser that is executing your test code

This enables Cypress to listen and modify the browser behavior at run time by manipulating DOM and altering Network requests and responses on the fly

Cypress open doors to New Kind of testing with Having ultimate control over your application (front and back)

**Cypress Browser Support:**

Chrome

Electron

Firefox & IE

**Cypress Components:**

Test Runner

Dash Board Service

**Course Outcome:**

By end of this course, You should be able to Automate any Web App using Cypress

You will understand how Cypress is Unique to build Non Flaky Stable Automation tests with the help of jQuery

You can mock network requests and responses with Cypress

Ability to Design Cypress framework from scratch with all the Testing standards

Integrate Cypress Test Framework to Jenkins for CI/CD

**Course Prerequisites:**

None for 90% of lectures.

Basics of API knowledge when dealing with API mocking topics. (10% lectures)

JavaScript Basics are taught in parallel when required with Cypress concepts

All Cypress files are spec File.

All Cypress files with .js

Cypress supports jasmen and mocha framework. But cypress recommends mocha framework.

Describe is like suite and it block contains the test in it and it is written under describe block.

Cy is global variable in cypress.

Fixtures is used to store test data. So it can be loaded directly in project.

Integration folder is used to write all our test cases.

Plugins are kind of listeners.

Support reusable method can be defined in support folder. All the methods can be directly available to use. We can directly call the method using it.

Node Modules - When we npm install.

Cypress Config – We had mentioned spec file location. So this file can be accessed at global level. All the properties are wrapped in e2e. Project level configuration can be changed by updating the config file.

|  |  |  |
| --- | --- | --- |
| Css Selector for cypress | | |
| **id** | tagname #idname |  |
| **classname** | .classname | .search-keyboard |
| **Custom with any attributes** | tagname[attribute=value] | input[type='search'] |
| **tagnames** |  | form input |

Cy.get to find element on webpage

Cy.

:visible and be used to count only visible elements on screen in cypress.

Parent child chaining

Checkbox

Locate checkbox

.check to check the box and verify by checking the value.

.uncheck to uncheck the box.

DropDown

Static Dropdown and Dynamic dropdown

.select method is used for dropdown

Cypress and Jquery

Cypress has no knowledge of child tab. So you cannot switch to child tab. But it can be done removing target Attribute from the html code. Invoke method is used from jquery.

Web Table using

:nth child

.next to go to next sibling in table

Pop up

Since there is no support of pop up in cypress.we use jquery function called Show method.

Child Tab

Prop method from jquery is used to get the attribute or value of property.

Frames

Cypress had no support for frames. But they recently did it.

Intall plugins npm install cypress -iframe

Hooks – Presented by mocha. It is same as test annotations in testNG.

All before() hook runs (Once)

Any beforeEach() hooks run

Test Run

Any AftereEach() hooks run

All After() hooks run (once)

Fixtures

Filename.json contains key value pair in it (“name”:”bob”,). Set up related part is written in before method before test cases.

Systax

Cy.fixture(“filename”)

This.data can be used in whole class.

Create custom command in command.js under support.

// -- This is a parent command --

// Cypress.Commands.add('login', (email, password) => { ... })

//

Cypress.Commands.add("selectProduct",(productName) => {

cy.get('h4.card-title').each(($el, index, $list) => {

if($el.text().includes(productName))

{

cy.get("button.btn.btn-info").eq(index).click()

}

})

})

cy.selectProduct("Nokia Edge")

cy.selectProduct("Blackberry")

This is how we can call that command in script

To pass array to object in example.json(datafile) we can put data inside [] and it will be considered as array of object.

Page Object Design Pattern

4 different pages in project (Basically 4 different class)

1. Home
2. Shop
3. Checkout
4. Products

If all the objects belonging to one page are found in 1 class.

Things to remember.

1. Create Class declare all page objects.
2. Export the class
3. Import the class
4. Create obj of that class

Make sure to mention return in method

Override existing behaviour

Global variables

Npx cypress run –spec (.js) location –headed –browser chrome –env url=’http://rahulshettyacademy.com’

885d98f9-6452-415c-98b7-9804fdcbbbf9