



Getting Started

>_ Install Cypress

```
npm install cypress --save-dev
```

>_ Open Cypress Test Runner

```
npx cypress open
```

>_ Write your first Cypress test

```
describe('My First Test', () => {
  it('Visits the Kitchen Sink', ()
=> {
    cy.visit('https://example.cypress.io')
  })
})
```

>_ Visit a URL

```
cy.visit('https://www.example.com')
```

Interactions

>_ Click an element

```
cy.get('#button').click()
```

>_ Type text into an input field

```
cy.get('#input').type('Hello, Cypress!')
```

>_ Select an option from a dropdown

```
cy.get('#select').select('option2')
```

>_ Check a checkbox

```
cy.get('#checkbox').check()
```

>_ Hover over an element

```
cy.get('#element').trigger('mouseover')
```

Waiting & Timeouts

>_ Wait for an element to be visible

```
cy.get('#element').should('be.visible')
```

>_ Wait for a specific time

```
cy.wait(2000)
```

>_ Set a custom timeout

```
Cypress.config('defaultCommandTimeout', 10000)
```

>_ Retry command until it passes

```
cy.get('#element').should('be.visible', { timeout: 10000 }).retry(3)
```

>_ Use `then` for asynchronous actions

```
cy.get('#element').then(($el) => {
  // Actions with element
})
```

Assertions

>_ Check element existence

```
cy.get('#element').should('exist')
```

>_ Check element visibility

```
cy.get('#element').should('be.visible')
```

>_ Check text content

```
cy.get('#element').should('contain', 'text')
```

>_ Check element attribute

```
cy.get('#element').should('have.attr', 'href', 'https://example.com')
```

>_ Assert element length

```
cy.get('li').should('have.length', 3)
```

Traversals

>_ Find children elements

```
cy.get('.parent').children('.child')
```

>_ Find parent element

```
cy.get('.child').parent('.parent')
```

>_ Find siblings

```
cy.get('.sibling1').siblings('.sibling2')
```

>_ Find next element

```
cy.get('.element').next('.nextElement')
```

>_ Find previous element

```
cy.get('.element').prev('.previousElement')
```

Network Requests & Aliases

>_ Intercept network requests

```
cy.intercept('GET', '/api/users').as('getUsers')
```

>_ Wait for intercepted requests

```
cy.wait('@getUsers')
```

>_ Stub a network request

```
cy.intercept('POST', '/api/users', { fixture: 'user' })
```

>_ Use fixtures for test data

```
cy.fixture('user').then((user) => {
  // Use user data
})
```

>_ Verify network response

```
cy.intercept('GET', '/api/users').as('getUsers');
cy.wait('@getUsers').its('body').should('have.length', 10);
```



Master Cypress Automation!

Become a Cypress expert and build reliable end-to-end tests for your web applications.

- Efficient Cypress setup and configuration
- Mastering Cypress commands for interactions & assertions
- Handling asynchronous operations and network requests
- Implementing data-driven testing with fixtures
- Comprehensive debugging and troubleshooting techniques

[Enroll now >>](#)

<https://ikalamtech.com/recommended-courses>

