### Assignment - 9 (Cypress)

**Q. Difference Between Selenium and Cypress Advantages of using cypress for web App Testing Explain the architecture of cypress.**

**Ans🡪**

|  |  |  |
| --- | --- | --- |
|  | **Selenium** | **Cypress** |
| **Language Support** | Selenium supports multiple programming languages like Java, C#, Python, Ruby, JavaScript, and Kotlin. | Cypress supports JavaScript, specifically designed for front-end developers. |
| **Browser Support** | It supports a wide range of browsers including Chrome, Firefox, Safari, Internet Explorer, and Edge. | It primarily supports Chrome-family browsers and Firefox. |
| **Framework** | Selenium WebDriver is a web automation framework that allows you to execute your tests against different browsers. | Cypress is an end-to-end testing framework built for modern web applications. |
| **Execution** | Selenium operates outside the browser and executes remote commands through the network. | Cypress operates inside the browser and directly interacts with the web application, which provides real-time feedback. |
| **Development Complexity** | Writing and maintaining tests in Selenium can be complex due to its asynchronous nature and dependency on WebDriver. | Cypress provides a simpler, more developer-friendly experience with built-in features like time travel, debugging, and real-time reloads. |
| **Community and Support** | Selenium has a large, active community and extensive documentation. | Cypress has a growing community and provides detailed documentation and guides. |

**Advantages of Using Cypress for Web App Testing**

1. **Real-Time Reloads:**
   * Tests run in real-time as you make changes, offering immediate feedback.
2. **Automatic Waiting:**
   * Cypress automatically waits for commands and assertions before moving on, eliminating the need for explicit waits.
3. **Time Travel:**
   * The ability to hover over each step of your test to see what happened at that moment in time.
4. **Network Traffic Control:**
   * Cypress can stub network requests and responses, making it easier to test edge cases and failures.
5. **Developer Experience:**
   * Built for developers with features like a powerful, debuggable browser environment and a simple API.
6. **Fast Execution:**
   * Cypress runs tests much faster due to its architecture, where it operates directly in the browser.
7. **Consistent Results:**
   * Since Cypress operates inside the browser, it produces more consistent test results.

**Architecture of Cypress**

1. **Test Runner:**
   * **Runs inside the browser**: Cypress test runner operates directly within the browser, running your tests as a part of the application under test.
   * **Interactive UI**: The test runner provides an interactive user interface that allows you to see commands and assertions as they execute in real-time.
2. **Node.js Server:**
   * **Backend Process**: A Node.js server runs alongside the browser to support features like network stubbing, file system access, and more.
   * **Proxy Layer**: Acts as a proxy server to capture and modify network requests and responses.
3. **Driver:**
   * **Execution Environment**: The driver is responsible for executing commands within the context of the browser, handling interactions with the DOM, and managing assertions.
4. **Reporter:**
   * **Real-Time Feedback**: The reporter provides real-time feedback on test execution, showing passed, failed, and pending tests, along with detailed error messages and stack traces.
5. **Dashboard Service:**
   * **Test Recording and Analytics**: Cypress provides a cloud-based dashboard service for recording test runs, viewing results, and analyzing trends.
6. **Plugins and APIs:**
   * **Extendable**: Cypress offers a plugin system and APIs to extend its functionality, integrate with other tools, and customize the testing environment.