**Assignment No. 6 –Cypress Automation**  
**Domain: QA (Quality Assurance)  
Organization: 10Pearls Pakistan**  
**Name: Zainab Arif**  
**Submission Date: 29th October 2025**

**Project Overview**

This assignment focused on automating UI testing for the **SauceDemo** website using the **Cypress framework**.  
The goal was to learn Cypress fundamentals, implement the Page Object Model (POM), and create reusable test cases through custom commands.

**Learning Objectives**

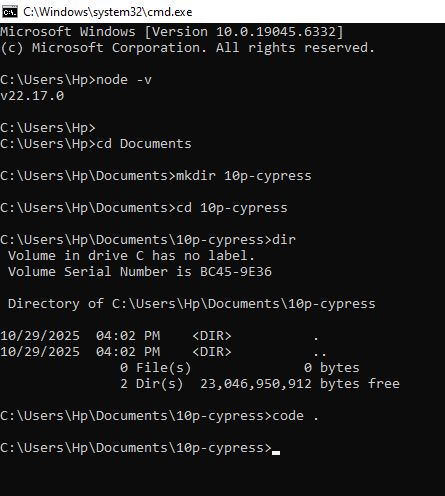
* Understand Cypress framework basics.
* Learn to automate login scenarios (success and failure).
* Implement reusable steps using **custom commands**.
* Apply **Page Object Model (POM)** for better test organization.
* Execute automated tests using the Cypress Test Runner.

**Tools & Technologies**

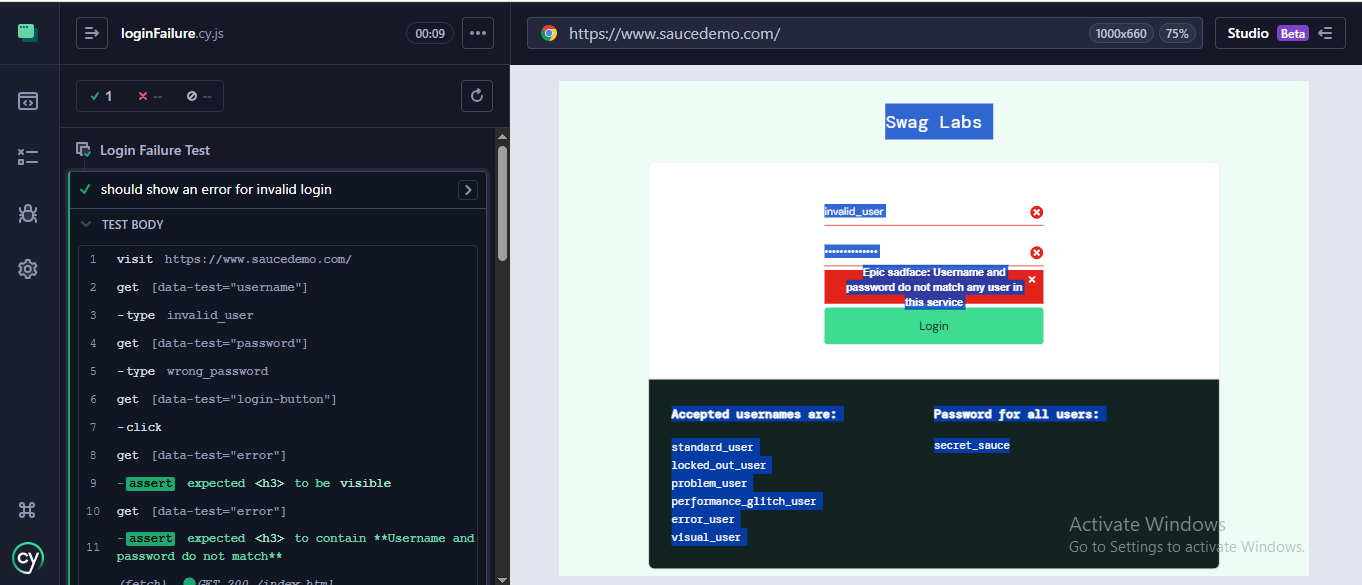
* **Framework:** Cypress
* **Language:** JavaScript
* **Node.js version:** 22.17.0
* **IDE:** Visual Studio Code
* **Version Control:** Git & GitHub

**Steps Performed**

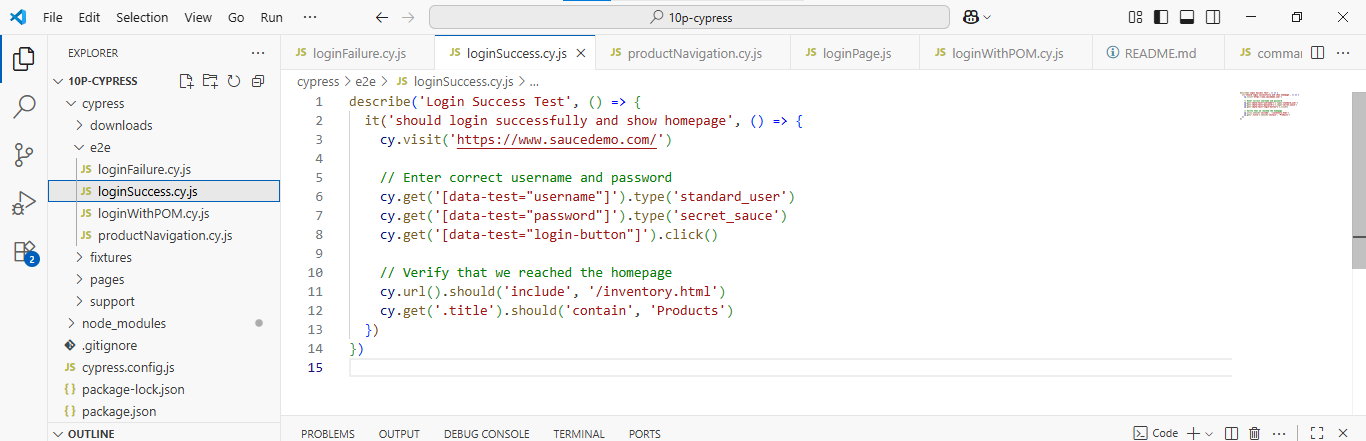
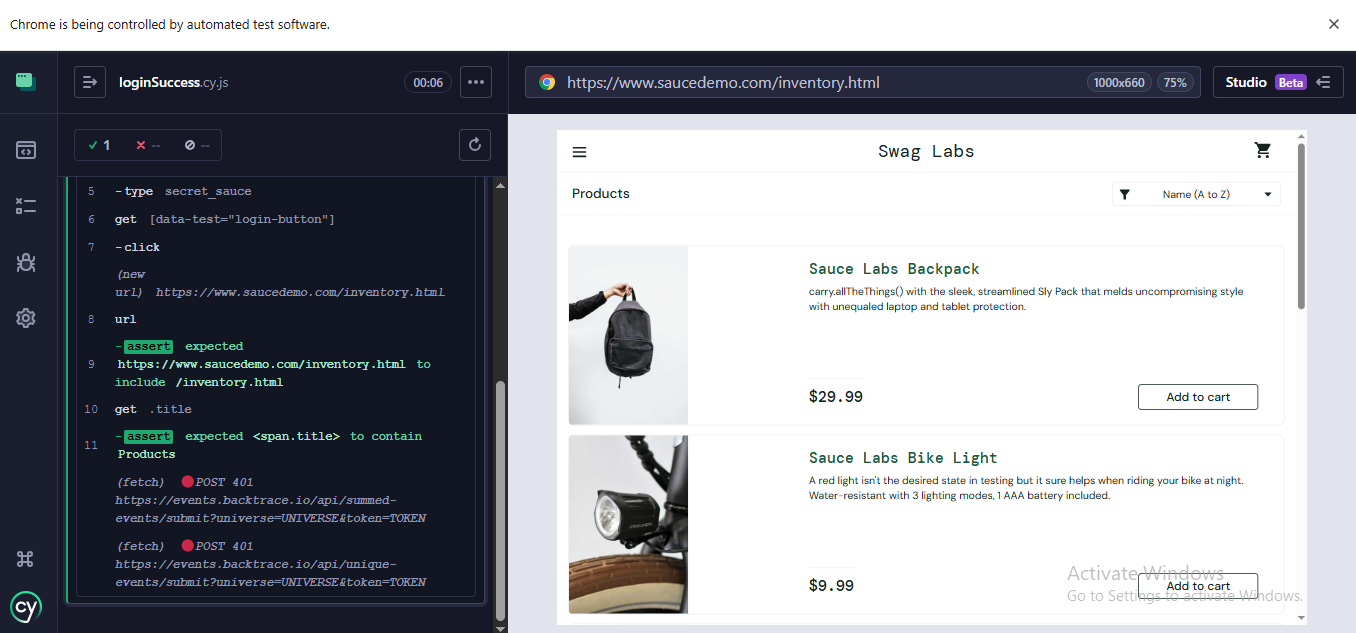
**Step 1: Setup Cypress Project**

* Verified Node.js installation (node -v).
* Created a new Cypress project folder.
* Initialized project using:
* npm init -y
* npm install cypress --save-dev
* npx cypress open
* Opened Cypress Test Runner and confirmed setup.  
    
  

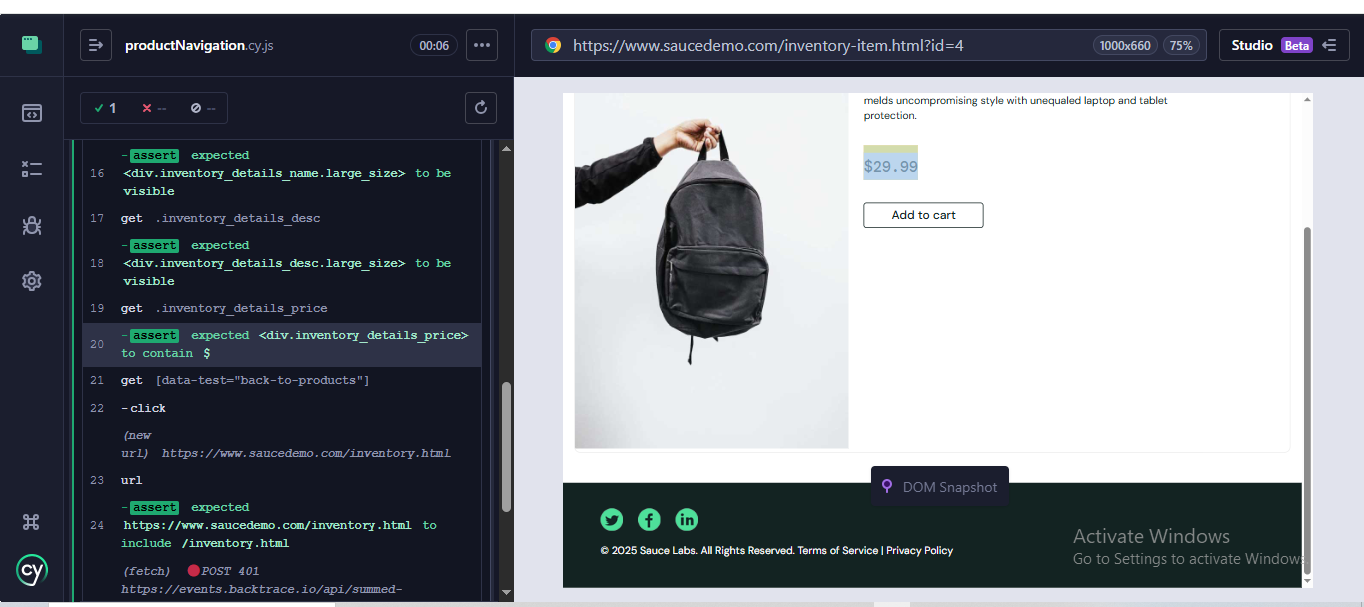
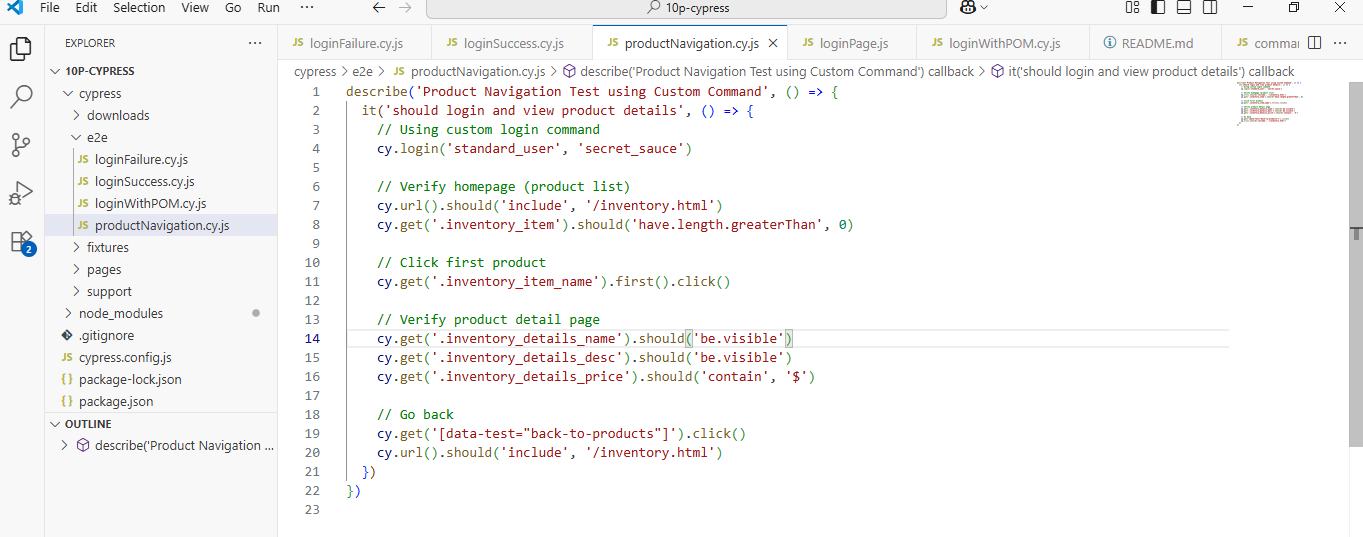
**Step 2: Automate Login Failure Scenario**

* Created a test file in cypress/e2e/loginFailure.cy.js
* Tested invalid login credentials for SauceDemo.
* Verified error message:
* Epic sadface: Username and password do not match any user in this service  
    
    
  

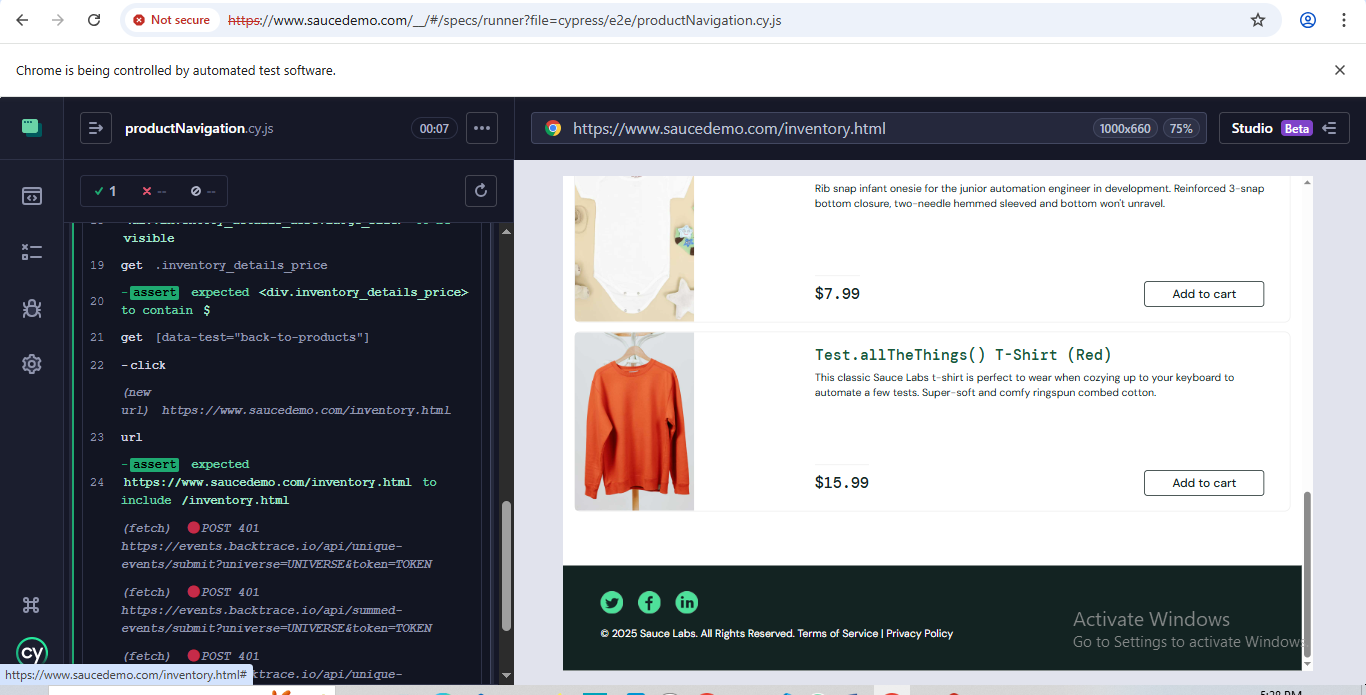
**Step 3: Automate Login Success Flow**

* Used valid credentials (standard\_user, password: secret\_sauce)
* Verified successful navigation to the **Products page**.
* Added assertions to confirm product visibility.  
    
  

**Step 4: Automate Product Navigation and Validation**

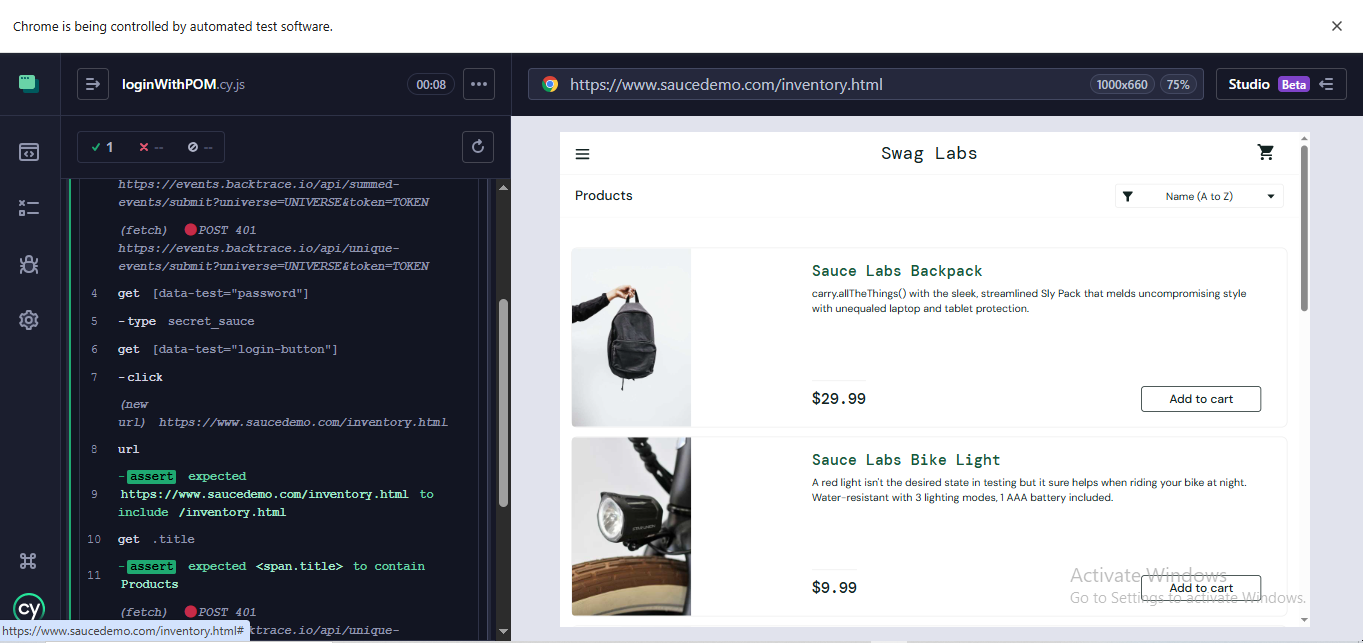
* Navigated through multiple products.
* Validated product titles, prices, and “Add to Cart” functionality.  
  

**Step 5: Create Custom Commands**

* Added reusable login command in cypress/support/commands.js:
* Cypress.Commands.add('login', (username, password) => {
* cy.get('[data-test="username"]').type(username)
* cy.get('[data-test="password"]').type(password)
* cy.get('[data-test="login-button"]').click()
* })
* Reused command across multiple tests to improve code efficiency.  
    
  

**Step 6: Apply POM (Page Object Model)**

* Created separate page files:
  + pages/LoginPage.js
  + pages/HomePage.js
* Used these pages to structure tests cleanly and improve maintainability.

**Step 7: Push Project to GitHub**

* Initialized Git and connected repository.
* Added .gitignore and README.md.
* Resolved merge conflicts and successfully pushed to GitHub:
* https://github.com/Zainab88804/Cypress---QA-Task

**Conclusion**

Through this project, I successfully:

* Installed and configured Cypress.
* Built automated test scripts for SauceDemo.
* Implemented best practices like reusable commands and POM structure.
* Gained hands-on experience in version control using Git and GitHub.