

# Writeup:-

**Github link:-** <https://github.com/akshansha9/JAVA-FSD.git>

## Project Objective:

The primary objective of this project is to leverage the provided website link and automate various functionalities within the web application. This includes creating a new project, setting up dependencies, developing automation and test scripts using Java, and executing the project using TestNG.

## Requirements:

To achieve the project objective, the following tools and frameworks must be employed:

1. Eclipse as the IDE: Utilized as the integrated development environment for coding and managing the project.
2. TestNG: Used for testing and organizing test cases, providing a robust testing framework.
3. Selenium WebDriver: Employed for automating interactions with the web application, ensuring seamless testing.

## Functionalities to be Automated:

### 1. Registration Form:

- Develop an automation script to navigate to the registration form.
- Fill in the required details and submit the form.
- Verify successful registration.

### 2. Login:

- Automate the login process with valid credentials.
- Handle invalid logins gracefully, displaying appropriate error messages in the console.

- Verify successful login.

### 3. Search Bar:

- Develop an automation script to interact with the search bar.
- Perform a search for a specific item.
- Verify that the search results match the expected outcome.

### 4. Add-to-Cart:

- Automate the process of adding an item to the cart.
- Verify that the item is successfully added to the cart.

### Handling Invalid Logins:

- Create scenarios for invalid logins and implement error handling.
- Display relevant error messages in the console for better debugging and analysis.

### Test Reports:

- Implement functionality to fetch test reports in HTML or Excel format.
- These reports should provide a comprehensive overview of test results, aiding in result analysis and debugging.

### Documentation:

#### 1. Project Setup:

- Create a new project in Eclipse.
- Include all dependencies in the classpath, ensuring a smooth execution environment.

#### 2. Automation and Test Scripts:

- Develop Java classes containing automation and test scripts for each functionality.
- Utilize Selenium WebDriver to interact with the web elements.

#### 3. TestNG Integration:

- Configure TestNG for organizing and executing test cases.
- Ensure seamless integration with the Eclipse project.

#### 4. Execution:

- Run the project using TestNG.
- Verify that all automated functionalities are executed successfully.

#### 5. Error Handling and Reports:

- Document the approach taken to handle invalid logins and display appropriate error messages.
- Showcase how test reports are fetched in HTML or Excel format.

=====