**ReadMe**

**Test automation assignment Enablon**

**Manual Test Cases**

**Positive Scenarios:-**

1. Insert and delete multiple data from Json file and validate all element are inserted and deleted

**Steps:**

1. Created one json file,I have created here size 5 Array, we can create any number of data size array, it is dynamic it is present in fixtures folder-testdata.json, we can add different type of data string in values array
2. Visit URL
3. Add all this data dynamically one by one in todos Textbox.
4. **Validate** data size from json values array with UI element in drop down, its dynamic.
5. Now Select all element by toggle all option, by selecting all it will strike all element
6. **Validate** **0 items** left option when user select all element
7. Now click on clear completed option
8. **Validate** all todo items element and footer should **disappear** from UI
9. When User Select radio botton for any String -:toggle item is not but footer is present in Active List but it is there in All and completed Option

**Steps:**

1. Visit URL
2. Enter only One string from json file in text box
3. Click on radio button
4. Click On **Active** link
5. **Validate** todo items element disappear from UI and footer should as it is.
6. Click on **Completed** and check status of todo drop down and footer
7. **Validate** Todo drop down and footer should visible
8. Click on **All** and check status of todo drop down and footer
9. **Validate** Todo drop down and footer should visible
10. When User doesn't Select radio botton for any String-: toggle view is not present in Completed but it is there for Active and All

**Steps:**

1. Visit URL
2. Enter only One string from json file in text box
3. Click on Completed
4. **Validate** todo items element disappear from UI and footer should as it is.
5. Click on **Active**
6. **Validate** Todo drop down and footer should visible
7. Click on **All** and check status of todo drop down and footer
8. **Validate** Todo drop down and footer should visible
9. Validate Cross icon and clear completed scenario

**Steps**:

1. Visit URL
2. Enter Single String from Json
3. Click on radio button of string in drop down
4. Click on **cross icon and validate scenario**
5. **Validate** Todo drop down and footer should not visible
6. Validate Clear Completed Scenario for New String
7. Visit URL
8. Enter Single String from json
9. Click on radio button of string in drop down
10. Click on Clear completed and Validate Result
11. **Validate** Todo drop down and footer should not visible

**Negative Scenarios:-**

1. **Validate Long String Scenario**

Steps:

1. Visit URL
2. Take Long String from Json file which has more than 109 character.
3. **Validate That Entered String and toggle drop down string should same**
4. **Validate empty space String Scenario**

Steps**:**

1. Visit URL
2. Take empty String from Json file
3. **Validate** Todo drop down and footer should not visible
4. **Validate Single Character Scenario**

Steps:

1. Visit Url
2. Take Single character from json file
3. **Validate todoList and footer not present**
4. **Enter Some JS Script and validate Scenario**

Steps:

1. Visit Url
2. Take JS script from json file
3. **Validate JS script data is present in dropdown**
4. **Enter duplicate string multiple time and validate Scenario**

Steps:

1. Visit Url
2. Take duplicate string from json time multiple time and add
3. **Validate duplicate String is accepted and showing in drop down**
4. **Note- I forget to add this scenario in automation script., Just I remember so added in notepad.**

**Automation:**

1. All above 8 test cases are automated with Cypress-JavaScript
2. Under cypress/e2e/ both positive and negative suite are there
3. Mocha Framework is used for this testing
4. Mochawesome reporting is used for reports
5. Reports are under report present
6. On failure User can see screenshot also

**To Run This Automation framework on any VM following things are require**

1. VS code require on Machine
2. Need to install nodejs
3. Import project from GIthub- <https://github.com/laxman7777/WKAssignment>
4. On Terminal do npm install cypress
5. To Run test Case from command line do npx cypress run