**Interacting with Forms using Selenium with C#**

Create Testpage.html

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8" />

<meta name="viewport" content="width=device-width, initial-scale=1.0" />

<title>Employee Registration Form</title>

<style>

body {

font-family: Arial, sans-serif;

background-color: #f4f4f4;

margin: 0;

padding: 0;

display: flex;

justify-content: center;

align-items: center;

height: 100vh;

}

.registration-form {

background-color: #fff;

padding: 20px;

border-radius: 8px;

box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);

width: 300px;

}

.registration-form h2 {

text-align: center;

margin-bottom: 20px;

}

.registration-form label {

display: block;

margin-bottom: 5px;

font-weight: bold;

}

.registration-form input[type="text"],

.registration-form input[type="email"],

.registration-form input[type="date"],

.registration-form select {

width: 100%;

padding: 8px;

margin-bottom: 15px;

border: 1px solid #ccc;

border-radius: 4px;

}

.registration-form fieldset {

border: 1px solid #ccc;

border-radius: 4px;

padding: 10px;

margin-bottom: 15px;

}

.registration-form fieldset legend {

font-weight: bold;

padding: 0 5px;

}

.registration-form input[type="radio"] {

margin-right: 5px;

}

.registration-form input[type="checkbox"] {

margin-right: 5px;

}

.registration-form button {

width: 100%;

padding: 10px;

background-color: #28a745;

color: #fff;

border: none;

border-radius: 4px;

cursor: pointer;

font-size: 16px;

}

.registration-form button:hover {

background-color: #218838;

}

</style>

<script>

function submitForm(event) {

event.preventDefault(); // Prevent actual form submission

// Hide the form and show a success message

document.getElementById("registration-form").style.display = "none";

document.getElementById("success-message").style.display = "block";

}

</script>

</head>

<body>

<div class="registration-form">

<h2>Employee Registration</h2>

<form id="registration-form" onsubmit="submitForm(event)">

<label for="name">Name:</label>

<input type="text" id="name" name="name" required />

<label for="email">Email:</label>

<input type="email" id="email" name="email" required />

<label for="doj">Date of Joining:</label>

<input type="date" id="doj" name="doj" required />

<fieldset>

<legend>Gender:</legend>

<label>

<input type="radio" name="gender" value="male" required />

Male

</label>

<label><input type="radio" name="gender" value="female" /> Female</label>

<label><input type="radio" name="gender" value="other" /> Other</label>

</fieldset>

<label for="city">City:</label>

<select id="city" name="city" required>

<option value="">Select City</option>

<option value="hyderabad">Hyderabad</option>

<option value="coimbatore">Coimbatore</option>

<option value="chennai">Chennai</option>

<option value="bangalore">Bangalore</option>

</select>

<label for="designation">Designation:</label>

<input type="text" id="designation" name="designation" required />

<label for="skills">Skills:</label>

<select id="skills" name="skills" multiple required>

<option value="testing">Testing</option>

<option value="java">Java</option>

<option value=".net">.NET</option>

<option value="cloud">Cloud</option>

</select>

<fieldset>

<legend>Hobbies:</legend>

<label>

<input type="checkbox"

name="hobbies"

value="Reading Books" />Reading Books

</label>

<label>

<input type="checkbox"

name="hobbies"

value="Playing Chess" />Playing Chess

</label>

<label>

<input type="checkbox"

name="hobbies"

value="Dancing" />Dancing

</label>

<label>

<input type="checkbox"

name="hobbies"

value="Playing Baseball" />Playing Baseball

</label>

</fieldset>

<button type="submit">Register</button>

</form>

<div id="success-message" class="success-message">

🎉 Thank you for registering! 🎉

</div>

</div>

</body>

</html>

Create Separate class in vs 2022 Nunit Test project

SeleniumCustomMethod (or) SeleniumHelper.cs

public static void Click(IWebElement locator)

{

locator.Click();

}

public static void Submit(IWebElement locator)

{

locator.Submit();

}

public static void EnterText(IWebElement locator, string text)

{

locator.Clear(); // clear the text from textbox

locator.SendKeys(text);

}

public static void SelectDropDownByText(IWebElement locator, string text)

{

SelectElement selectElement = new SelectElement(locator);

selectElement.SelectByText(text);

}

public static void SelectDropDownByValue(IWebElement locator, string value)

{

SelectElement selectElement = new SelectElement(locator);

selectElement.SelectByValue(value);

}

public static void MultiSelectElements(IWebElement locator, string[] values)

{

SelectElement multiSelect = new SelectElement(locator);

foreach (var value in values)

{

multiSelect.SelectByValue(value);

}

}

public static List<string> GetAllSelectedLists(IWebElement locator)

{

List<string> options = new List<string>();

SelectElement multiSelect = new SelectElement(locator);

IList<IWebElement> selectedOption = multiSelect.AllSelectedOptions;

foreach (var option in selectedOption)

{

options.Add(option.Text);

}

return options;

}

public static List<string> GetAllCheckedCheckboxes(IWebDriver driver, IWebElement locator)

{

List<string> selectedValues = new List<string>();

var checkboxes =driver.FindElements((By)locator); ;

foreach (var checkbox in checkboxes)

{

if (checkbox.Selected) // If the checkbox is checked

{

selectedValues.Add(checkbox.GetAttribute("value"));

}

}

return selectedValues;

}

Add another class HtmlFormTest.cs

IWebDriver driver;

public void Setup()

{

driver=new ChromeDriver();

}

[Test]

public void WorkingWithAdvancedControls()

{

IWebDriver driver = new ChromeDriver();

driver.Manage().Timeouts().ImplicitWait = TimeSpan.FromSeconds(2);

driver.Navigate().GoToUrl("file:///C:/Personal/cmr\_capgemini/Testing/Selenium-with-C-Notes/Selenium/Testpage.html");

FormHelper.EnterText(driver.FindElement(By.Id("name")), "Geetha Eswaramurthi");

FormHelper.EnterText(driver.FindElement(By.Id("email")), "geethaeswaramurthi@outlook.com");

FormHelper.EnterText(driver.FindElement(By.Id("doj")), "8-3-2020");

FormHelper.Click(driver.FindElement(By.CssSelector("input[name='gender'][value='female']")));

FormHelper.SelectDropDownByText(driver.FindElement(By.Id("city")), "Coimbatore");

FormHelper.EnterText(driver.FindElement(By.Id("designation")), "Senior Product Engineer");

FormHelper.MultiSelectElements(driver.FindElement(By.Id("skills")), ["testing", "cloud"]);

var getSelectedOptions = FormHelper.GetAllSelectedLists(driver.FindElement(By.Id("skills")));

getSelectedOptions.ForEach(Console.WriteLine);

FormHelper.Click(driver.FindElement(By.CssSelector("input[name='hobbies'][value='Reading Books']")));

FormHelper.Click(driver.FindElement(By.CssSelector("input[name='hobbies'][value='Playing Baseball']")));

var getSelectedHobbies = FormHelper.GetAllCheckedCheckboxes(driver, driver.FindElement(By.Name("hobbies")));

getSelectedHobbies.ForEach(Console.WriteLine);

driver.Manage().Timeouts().ImplicitWait = TimeSpan.FromSeconds(2);

// Clicking the Register Button

FormHelper.Click(driver.FindElement(By.CssSelector("button[type='submit']")));

// Validation: Ensure form submission redirects to a success page or displays a confirmation message

Thread.Sleep(2000); // Wait for form submission response

bool isSubmissionSuccessful = driver.PageSource.Contains("Thank you for registering") || driver.Url.Contains("success");

Assert.IsTrue(isSubmissionSuccessful, "Form submission failed.");

}

[TearDown]

public void TearDown()

{

driver.Close();

driver.Quit();

}