# **Day 2\_Selenium\_Practice2**

## 1. Scenario 1

using NUnit.Framework;

using OpenQA.Selenium;

using OpenQA.Selenium.Chrome;

using OpenQA.Selenium.Support.UI;

using System;

using System.Collections.Generic;

using System.Threading;

namespace Automation\_1

{

[TestFixture]

public class Tests

{

protected WebDriverWait \_wait;

protected IWebDriver \_driver;

[SetUp]

public void Setup()

{

\_driver = new ChromeDriver(AppDomain.CurrentDomain.BaseDirectory);

\_wait = new WebDriverWait(\_driver, TimeSpan.FromSeconds(10));

}

[Test]

public void BrowserCommandTests()

{

//Browser commands

\_driver.Navigate().GoToUrl("http://automationpractice.com/index.php");

\_driver.Manage().Window.Maximize();

Console.WriteLine("Web open successfully!");

IWebElement click = \_driver.FindElement(By.XPath("//\*[@id=\"contact-link\"]/a"));

click.Click();

String Title = \_driver.Title;

string expectedTitle = "CUSTOMER SERVICE - CONTACT US";

Console.WriteLine($"Title of the form is: "+expectedTitle);

\_driver.Navigate().Back();

Console.WriteLine($"Title of the page is: {\_driver.Title}");

\_driver.Navigate().Forward();

\_driver.Close();

}

## **2. Implicit wait and Explicit wait**

|  |  |
| --- | --- |
| Implicit Wait | Explicit Wait |
| Implicit Wait tells the Webdriver maximum wait time when searching for elements before throwing exception | Explicit Wait tells the Webdriver to halt the execution until a particular condition is met or the maximum time has passed |
| Implicit Wait is global, it is applicable to all the web elements in the script | Explicit Wait time is applicable only to those web elements which are specified by the user |
| Since Implicit Wait applies to all web elements, users do not specify “Expected Conditions” on the element to be located | Explicit Wait requires the expected conditions to be specified for elements, like “Wait untile element is enabled” |

## **3. Write wait methods using explicit wait method to wait:**

• Wait for an element is displayed

public IWebElement ReturnWebElement(By by)

{

IWebElement e = \_driver.FindElement(by);

\_wait.Until(SeleniumExtras.WaitHelpers.

ExpectedConditions.ElementIsVisible(by));

return e;

}

• Wait for an element is clickable

public IWebElement ReturnWebElement(By by)

{

IWebElement e = \_driver.FindElement(by);

\_wait.Until(SeleniumExtras.WaitHelpers.

ExpectedConditions.ElementToBeClickable(by));

return e;

}

• Wait for an element is selected

public IWebElement ReturnWebElement(By by)

{

IWebElement e = \_driver.FindElement(by);

\_wait.Until(SeleniumExtras.WaitHelpers.

ExpectedConditions.ElementToBeSelected(by));

return e;

}