Advanced Selenium in C#

1. **Introduction to Automation Testing**
    - What automation testing?
    - Automation vs Manual testing
    - Advantage and disadvantage of automation testing
    - **Metrics for automation testing**
        - Technical behaviour of the project
        - Complexity of the project
        - Stability of the project
        - Test data Size
        - Application Size
        - Reusability if the script
        - Execution platforms of the project
    - **Available tools for automation testing**
        - Selenium
        - UFT(Unified Functional Testing)
2. **Selenium Introduction**
    - Selenium IDE
    - Selenium Webdriver
    - Selenium Grid
3. **Setup Selenium Webdriver in Visual Studio**
    - Downloading Selenium Webdriver
    - Import webdriver to C# project
    - Running your first selenium test using C#
4. **Create first project**
    - Click "New project" → Console Application
    - Give project Name, Change Location, and give solution Name
    - Solution is a folder that stores number of projects
    - Inside your project Change the Class name appropriately
5. **Installing and configuring selenium webDriver**
    - From Nuget or Extensions

   - Search by "Selenium.webdriver" and install
   - **First test case:**
     - Create a new class in your project
     - **Add the following selenium namespaces**
       1. using OpenQA.Selenium;
       2. using OpenQA.Selenium.Firefox;
     - **Add the following in main method**
       1. IWebDriver driver = new FirefoxDriver();
       2. driver.Url = "http.qatraining.com";
     - Run the test by clicking "Start" button
   - Tools → Windows → Test Explorer
   - Setup Nunit Framework and NunitTestAddapter
6. **Running Selenium webdriver in multi browsers:**
   - Firefox, Chrome
   - IE, Safari
   - HtmlUnitDriver
7. **What is Nunit Framework**
   - Install Nunit Framework from Nuget or Extensions
   - Install Nunit Test Adapter for test report
   - **Create your first Nunit class**
     - Create a new class in you project
     - **Add three methods and annotate with Nunit as follows**
       1. [SetUP] to run before each test
       2. [Test] to run actual test
       3. [TearDown] to run after each test
8. IWebDriver Introduction
9. **IWebDriver Commands**
   - Close: to close current window only
   - Quite : to close all windows opened by webdriver
   - currentWindowHandle
   - Pagesource, Title, Url
   - **findeElement/s By**
     - Id, Name, Tagname

        - Absolute and relative XPath
        - clssName, cssSelector
        - LinkText and PartialLinkText
    - **SwitchTo**
        - ActiveElement
        - **Alert**
            1. Accept, Dismiss
            2. SendKeys, Text
        - DefaultContent
        - Frame, ParentFrame
        - Window
    - **Navigate**
        - Back, Forward
        - goToUrl, Refresh
    - Managing Web pages

10. **WebElement Commands**
    - Clear, Click
    - Submit, Text, Tagname
    - Displayed, Enabled
    - Selected, SendKeys
    - GetAttribute, GetCssValue, GetHashCode
    - GetType, Location
    - Size : returns Dimensions.x and dimensions.y
11. Checkbox and Radio Button Operations
12. **Using selected reference**
    - IList<IWebElement> radioButton= driver.FindElements(By.Name("Name here"))
    - Bool check= radioButton.ElementAt(0).Selected;
13. **Using Value reference**
    - Using for loop of radioButton.Count
    - Using radioButton.ElementAt(i).GetAttribute(Value);
    - If value matches our value
    - Then take click action.
14. **Using CssSelector is very simple way**

   - driver.FindElement(By.CssSelector(input[value="your value"]).Click()

15. DropDown and Multiple Select Operation
16. **Using Select Class of Selenium**
    - SelectElement oSelect = new SelectElement(driver.FindElement(By.Id("Country")));
    - **SelectElement Options:**
        - AllSelectedOptons
        - Select/DeselectAll
        - Select/DeselectByID
        - Select/DeselectByIndex
        - Select/DeselectByValue
        - Select/DeselectByText
        - IsMultiple
        - Optoins: IList<IWebElement> elementCount= oSelect.Options
        - SelectedOption
17. WebDriver Wait Elements
18. Implicit wait
19. **Explicit wait**
    - **Using lambda Expression**
        - Accepts one input only
    - **Using Func Delegate**
        - Accepts numbers of input as parameter
20. Selenium Automation Frameworks
21. **Setup the platform for automation**
    - Install visual studio, Selenium webDriver, Nunit, Nunit Test Adapter
    - Create your project, create a new test folder
    - Write your first test class using Nunit annotation
22. **Using Page Factory**
    - **Advantage**
        - Easy to maintain
        - Reduce code redundancy
        - More readable
        - Reusable

        - Reliable
    - @FindBy annotation
    - InitElements method: to instantiate a class given to it
    - **PageFactory NameSpace**
        - How to apply page factory in selenium
        - Create home page object
        - Create login page object
        - Write test for LogIn functionality

23. **Logging using Log4Net**
    - Introduction to logging
    - Download and import Log4Net
    - Writing test cases with Log4Net
    - Log4Net manager
    - Log4Net appenders
    - Log4Net logger
24. **PageFactory CatcheLookup**
    - How to use PageFactory CatcheLookup
    - PageFactoryCatchLookup NameSpace
    - **How to implement PageFactoryCatchLookup in Frameworks**
        - Advnatages of PageFactory CatcheLookup
25. **Way to optimize page object model**
    - Use constructor to initialize elements of each page
    - Bind elements to methods in the pabe object class
26. **Encapsulating selenium page objects**
    - Declare variables a privat
    - Use public get and set methods to modify variables
27. **Using configuration manager for manage and reading configurations**
    - **Using ConfiguraionManagerClass**
        - Reading from connectionStrings
        - Reading from AppSettings
        - Reading from External configuration file
        - Reading configuration String using ConnectionManager
    - Implementing configurations in automation frameworks

28. **Data Driven Testing**
    - Using Datapools
    - Using Excell files
    - Using ADO objects
    - Using CSV files
    - Using ODBC
29. Patterns for Browser/ webDriver factory
30. Page generator
31. webElement extension method
32. **Special tips:**
    - Handling file upload and download in different ways
    - Automating to test all links
    - Checking logo and address position
    - Work with https and proxy authentication
    - Handling windows alerts using autoIT