1. Difference between driver.close() and driver.quit()?

Selenium web driver provides driver.close() which is useful in closing current browser/window which is currently open by Selenium WebDriver.

Selenium WebDriver providing another method to close the browsers same as driver.close() method of selenium web driver, driver.quit() method perform same action as driver.close() but with small difference that is it will close all the browsers/window which is open by selenium Web driver.

1. Different types of waits or synchronization in selenium web driver?

Synchronization/waits can be achieved in many different ways:

1.sleep method

2.page load timeout

3.script timeout

4.implicit wait

5.explicit wait

Sleep() command:

The command halts the execution of the script for specified amount of milliseconds.

Example: Thread.sleep(2000);

Page Load timeout:

We can set the amount of time to wait for a page load to complete before throwing an error.

Example: driver.manage().timeouts().pageLoadTimeout(20,TimeUnit.SECONDS);

The webdriver instance waits for 20 seconds for every page to get loaded before throwing an exception. If the page is not loaded in 20 seconds then it throws TimedOutException at run time.

Script timeout:

We can set the amount of time to wait for an asynchronous script to finish execution before throwing any error.

Example: driver.manage().timeouts().setScriptTimeout(20,TimeUnit.Seconds);

The webdriver instance waits for 20 seconds for every asynchronous script to get executed on the web page before throwing an exception.

Implicit wait:

Implicit wait is a mechanism, which is used to notify webdriver instance to wait for specific time, if any element is readily not available on the webpage

Example: driver.manage().timeouts().implicitlyWait(10, TimeUnit.SECONDS);

The above line once added at the beginning of the automation script, the WebDriver will wait for 10 SECONDS for any element on the web page to be loaded before throwing any exception.

Once set, the timing will be set **for the life of the WebDriver instance** until and unless it is changed again. Meaning, using implicit wait command, we can say the **WebDriver instance to wait for specified amount time** before throwing NoSuchElementException, if it does not find any element on the web page immediately.

The synchronization mechanism is **not associated with any particular web element**, so it will be **applicable for all the elements that our script interacts with**, it is not highly appreciated.

Explicit wait:

Explicit wait is a technique used to advise the **WebDriver instance to stand by** the **execution** till the time the **certain condition is met** or **the maximum wait time is elapsed**.

Example:

WebElement element = driver.findElement(By.linkText(“Home”));WebDriverWait wait = new WebDriverWait(driver, 20);

wait.until(ExpectedConditions.visibilityOf(element));

WebDriverWait wait = new WebDriverWait(driver, 20);

The above line sets maximum time the WebDriver instance will wait for the condition to become true.

wait.until(ExpectedConditions.visibilityOf(element));

The above line asks WebDriver instance **to wait until the element is visible on the web page**.

1. How to assign the value to textbox other than sendKeys method?

The best alternative is to pass web element as an argument to the javascript

**Example:**

package junit;

import static org.junit.Assert.\*;

import org.junit.After;

import org.junit.Before;

import org.junit.Test;

import org.openqa.selenium.By;

import org.openqa.selenium.JavascriptExecutor;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.WebElement;

import org.openqa.selenium.firefox.FirefoxDriver;

public class testindeedsite {

static WebDriver driver;

public String baseUrl;

@Before

public void setUp()

{

System.setProperty("webdriver.gecko.driver","C:\\Users\\Sivachaitanya\\Downloads\\Compressed\\geckodriver-v0.13.0-win64\\geckodriver.exe");

driver = new FirefoxDriver();

baseUrl="http://www.indeed.com/";

}

@Test

public void testIndeed() throws InterruptedException

{

driver.get(baseUrl);

Thread.sleep(3000);

WebElement searchbox = driver.findElement(By.xpath("//input[@id='what']"));

JavascriptExecutor myExecutor = ((JavascriptExecutor) driver);

myExecutor.executeScript("arguments[0].value='selenium';", searchbox);

Thread.sleep(3000);

driver.findElement(By.id("where")).clear();

Thread.sleep(3000);

driver.findElement(By.id("where")).sendKeys("texas");

Thread.sleep(3000);

driver.findElement(By.id("fj")).click();

Thread.sleep(3000);

String expectedTitle=driver.getTitle();

String actualTitle=driver.getTitle();

assertEquals("error:job title is not matched",actualTitle,expectedTitle);

}

@After

public void tearDown()

{

driver.quit();

}

}