1. Write a test case to capture the screenshots with WebDriver.

import java.io.File;

import org.apache.commons.io.FileUtils;

import org.openqa.selenium.OutputType;

import org.openqa.selenium.TakesScreenshot;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.firefox.FirefoxDriver;

import org.testng.annotations.Test;

public class BStackTakeScreenshot

{

public void testBStackTakeScreenShot() throws Exception

{

WebDriver driver ;

System.setProperty("webdriver. firefox.marionette","c:\\geckodriver.exe");

driver = new FirefoxDriver();

driver. Get("https://www.browserstack.com");

this.takeSnapShot(driver, "c://test.png") ;

}

public static void takeSnapShot(WebDriver webdriver, String fileWithPath) throws Exception

{

TakesScreenshot scrShot =((TakesScreenshot)webdriver);

File SrcFile=scrShot.getScreenshotAs(OutputType.FILE);

File DestFile=new File(fileWithPath);

FileUtils.copyFile(SrcFile, DestFile);

}

}

2. Click on an element which is in iFrame.

var iframe = document.getElementById("myFrame");

var elmnt=iframe.contentWindow.document.getElementsByTagName("H1")[0];

elmnt.style.display = "none";

3. Find out the broken links on a web page.

import java.io.IOException;

import java.net.HttpURLConnection;

import java.net.MalformedURLException;

import java.net.URL.

import java.util.Iterator;

import java.util.List;

import org.openqa.selenium.By;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.WebElement;

import org.openqa.selenium.chrome.ChromeDriver;

public class BrokenLinks

{

private static WebDriver driver = null;

public static void main (String [] args)

{

String homePage = "http://www.zlti.com";

String url = "";

HttpURLConnection huc = null;

int respCode = 200;

driver = new ChromeDriver();

driver.manage().window().maximize();

driver.get(homePage);

List<WebElement> links = driver.findElements(By.tagName("a"));

Iterator<WebElement> it = links.iterator();

while(it.hasNext())

{

url = it.next().getAttribute("href");

System.out.println(url);

if (url == null || url.isEmpty()){

System.out.println("URL is either not configured for anchor tag or it is empty");

continue;

}

if(!url.startsWith(homePage))

{

System.out.println("URL belongs to another domain, skipping it.");

continue;

}

try {

huc = (HttpURLConnection)(new URL(url).openConnection());

huc.setRequestMethod("HEAD");

huc.connect();

respCode = huc.getResponseCode();

if(respCode >= 400)

{

System.out.println(url+" is a broken link");

}

else {

System.out.println(url+" is a valid link");

}

} catch (MalformedURLException e) {

e.printStackTrace();

} catch (IOException e) {

e.printStackTrace();

}

}

driver.quit();

}

}

4. Implicit and Explicit wait commands.

Implicit wait

import java.util.concurrent.TimeUnit;

import org.openqa.selenium.By;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.WebElement;

import org.openqa.selenium.chrome.ChromeDriver;

import org.openqa.selenium.support.ui.ExpectedConditions;

import org.openqa.selenium.support.ui.WebDriverWait;

public class ImplicitWait

{

public static void main (String[] args) throws InterruptedException

{

System.setProperty("webdriver. chrome.driver", "C:Selenium-java-

javaTpointchromedriver\_win32chromedriver.exe");

WebDriver driver = new ChromeDriver();

driver.manage().window().maximize();

driver.manage().deleteAllCookies();

driver.manage().timeouts().pageLoadTimeout(40,

TimeUnit.SECONDS);

driver. manage (). timeouts ().implicitlyWait(20, TimeUnit.SECONDS);

driver.get("https://login.google.com/");

driver.findElement(By.xpath("//input[@id='login-

username']")). sendKeys("JavaTpoint.com");

Thread.sleep(1000);

driver. findElement(By.xpath("//input[@id='login-signin']")).click();

}

}

Explicit wait

import java.util.concurrent.TimeUnit;

import org.openqa.selenium.By;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.WebElement;

import org.openqa.selenium.chrome.ChromeDriver;

import org.openqa.selenium.support.ui.ExpectedConditions;

import org.openqa.selenium.support.ui.WebDriverWait;

public class Locators

{

public static void main (String [] args) throws InterruptedException

{

System.setProperty("webdriver. chrome.driver", "C:Selenium-java-

javatpointchromedriver\_win32chromedriver.exe");

WebDriver driver = new ChromeDriver();

driver.manage().window().maximize();

driver.manage().deleteAllCookies();

driver.manage().timeouts().pageLoadTimeout(40,

TimeUnit.SECONDS);

driver. manage (). timeouts ().implicitlyWait(30, TimeUnit.SECONDS);

driver.get("https://www.facebook.com/");

WebElement firstname= driver. findElement(By.name("firstname"));

WebElement lastname= driver.findElement(By.name("lastname"));

sendKeys(driver, firstname, 10, "Edureka");

sendKeys(driver, lastname, 20, "Edureka");

WebElement forgotAccount=

driver.findElement(By.linkText("Forgotten account?"));

clickOn(driver,forgotAccount, 10);

driver. manage (). timeouts ().implicitlyWait(30, TimeUnit.SECONDS);

}

public static void sendKeys (WebDriver driver1, WebElement element,

int timeout, String value)

{

new WebDriverWait(driver1,

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

element.sendKeys(value);

}

public static void clickOn (WebDriver driver1, WebElement element,

int timeout)

{

new WebDriverWait(driver1,

timeout). until (ExpectedConditions.elementToBeClickable(element));

element.click();

}

}

5. Action class with the following operations

Keyboard key press event

Pressing enter button on the keyboard

ClickAndHold event,

Drag and Drop MoveToElement,

Mouse Hover Event Double Click event

public static void main (String[] args)

{

String baseUrl = "http://www.facebook.com/";

WebDriver driver = new FirefoxDriver();

driver.get(baseUrl);

WebElement txtUsername = driver.findElement(By.id("email"));

Actions builder = new Actions(driver);

Action seriesOfActions = builder

.moveToElement(txtUsername)

. click()

.keyDown(txtUsername, Keys.SHIFT)

.sendKeys(txtUsername, "hello")

.keyUp(txtUsername, Keys.SHIFT)

.doubleClick(txtUsername)

.contextClick()

. build();

seriesOfActions.perform() ;

}

6.Web Table operations

Get row count

Get data from a specific cell

Dynamic web table handling

Import unittest

import time

from selenium import webdriver

from selenium.webdriver.support.select import Select

from selenium.webdriver.common.by import By

from selenium.webdriver.support.ui import WebDriverWait

from selenium.webdriver.support import expected\_conditions as EC

test\_url = "https://www.w3schools.com/html/html\_tables.asp"

class WebTableTest(unittest.TestCase):

def setUp(self):

self.driver = webdriver.Chrome()

self.driver.maximize\_window()

def test\_1\_get\_num\_rows\_(self):

driver = self.driver

driver.get(test\_url)

WebDriverWait(driver, 60).until(EC.presence\_of\_element\_located((By.CLASS\_NAME, "w3-example")))

num\_rows = len (driver.find\_elements\_by\_xpath("//\*[@id='customers']/tbody/tr"))

print("Rows in table are " + repr(num\_rows))

def test\_2\_get\_num\_cols\_(self):

driver = self.driver

driver.get(test\_url)

WebDriverWait(driver, 60).until(EC.presence\_of\_element\_located((By.CLASS\_NAME, "w3-example")))

# num\_cols = len (driver.find\_elements\_by\_xpath("//\*[@id='customers']/tbody/tr/th"))

num\_cols = len (driver.find\_elements\_by\_xpath("//\*[@id='customers']/tbody/tr[2]/td"))

print("Columns in table are " + repr(num\_cols))

def tearDown(self):

self.driver.close()

self.driver.quit()

if \_\_name\_\_ == "\_\_main\_\_":

unittest.main()

7. Handling Ajax Auto suggestion.

import org.openqa.selenium.By;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.WebElement;

import org.openqa.selenium.chrome.ChromeDriver;

import org.openqa.selenium.support.ui.ExpectedConditions;

import org.openqa.selenium.support.ui.WebDriverWait;

import org.testng.Assert;

import org.testng.annotations.BeforeClass;

import org.testng.annotations.Test;

public class Ajaxdemo

{

private String URL = "http://demo.guru99.com/test/ajax.html";

WebDriver driver;

WebDriverWait wait;

public void setUp()

{

System.setProperty("webdriver.chrome.driver",".\\chromedriver.exe");

driver = new ChromeDriver();

driver.manage().window().maximize();

driver.navigate().to(URL);

}

public void test\_AjaxExample()

{

By container = By.cssSelector(".container");

wait = new WebDriverWait(driver, 5);

wait.until(ExpectedConditions.presenceOfElementLocated(container));

WebElement noTextElement = driver.findElement(By.className("radiobutton"));

String textBefore = noTextElement.getText().trim();

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

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

WebElement TextElement = driver.findElement(By.className("radiobutton"));

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

String textAfter = TextElement.getText().trim();

Assert.assertNotEquals(textBefore, textAfter);

System.out.println("Ajax Call Performed");

String expectedText = "Radio button is checked and it's value is Yes";

Assert.assertEquals(textAfter, expectedText);

driver.close();

}

}

8. Select a specific date from a calendar.

import java.util.List;

import java.util.concurrent.TimeUnit;

import org.openqa.selenium.By;

import org.openqa.selenium.Keys;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.WebElement;

import org.openqa.selenium.chrome.ChromeDriver;

import org.testng.annotations.Test;

public class DateTimePicker

{

public void dateTimePicker()

{

System.setProperty("webdriver.chrome.driver", "chromedriver.exe");

WebDriver driver = new ChromeDriver();

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

driver.get("http://demo.guru99.com/test/");

WebElement dateBox = driver.findElement(By.xpath("//form//input[@name='bdaytime']"));

dateBox.sendKeys("09252013");

dateBox.sendKeys(Keys.TAB);

dateBox.sendKeys("0245PM");

}

}