JAT Task 11

1.

package Task11;

import java.time.Duration;

import java.util.Set;

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 ques1frames {

static WebDriver *driver*;

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

*handleMultipleWindows*();

}

public static void handleMultipleWindows() throws InterruptedException {

*driver* = new ChromeDriver();

*driver*.get("https://the-internet.herokuapp.com/windows");

*explicitWaitForElement*(By.*xpath*("//a[contains(text(),'Click Here')]")).click();

Thread.*sleep*(2000);

String parentWidnow = *driver*.getWindowHandle();

Set<String> allWindows = *driver*.getWindowHandles();

for (String windowHandle : allWindows) {

if (!windowHandle.equals(parentWidnow)) {

*driver*.switchTo().window(windowHandle);

String titleOfNewWindow = *driver*.getTitle();

if (titleOfNewWindow.equals("New Window")) {

System.***out***.println("titleOfNewWindow is :: " + titleOfNewWindow);

} else {

System.***out***.println("title is :: " + titleOfNewWindow);

System.***out***.println("User not switched to New Window");

}

}

}

*driver*.close();

System.***out***.println("New window closed");

*driver*.switchTo().window(parentWidnow);

String titleOfParentWindow = *driver*.getTitle();

System.***out***.println("titleOfParentWindow is::"+titleOfParentWindow);

*driver*.quit();

}

public static WebElement explicitWaitForElement(By byElement) {

WebDriverWait wait = new WebDriverWait(*driver*, Duration.*ofSeconds*(30));

WebElement myElement = wait.until(ExpectedConditions.*visibilityOfElementLocated*(byElement));

return myElement;

}

}

Output:

titleOfNewWindow is :: New Window

New window closed

titleOfParentWindow is :: The Internet

2.Nested frames

package Task11;

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 ques2nestedframes {

static WebDriver *driver*;

public static void main(String[] args) {

*launchwebsite*("http://the-internet.herokuapp.com/nested\_frames");

*switchingframes*();

*driver*.quit();

}

public static void launchwebsite(String urlvalue) {

*driver* = new ChromeDriver();

*driver*.get(urlvalue);

*waitForTime*(2000);

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

}

public static void switchingframes() {

*driver*.switchTo().frame("frame-top");

List<WebElement> frameList = *driver*.findElements(By.*xpath*("//frame"));

int numberOfFrames = frameList.size();

if (numberOfFrames == 3) {

System.***out***.println("No of frames present in top frame:: 3");

} else {

System.***out***.println("No of frames present in top frame:: " + numberOfFrames);

}

*driver*.switchTo().frame("frame-left");

String leftText = *driver*.getPageSource();

if (leftText.contains("LEFT")) {

System.***out***.println("Left frame has the text 'LEFT'.");

} else {

System.***out***.println("Left frame does not have the text 'LEFT'.");

}

*driver*.switchTo().parentFrame();

*driver*.switchTo().frame("frame-middle");

String middleText = *driver*.getPageSource();

if (middleText.contains("MIDDLE")) {

System.***out***.println("Middle frame has the text 'MIDDLE'.");

} else {

System.***out***.println("Middle frame does not have the text 'MIDDLE'.");

}

*driver*.switchTo().parentFrame();

*driver*.switchTo().frame("frame-right");

String rightText = *driver*.getPageSource();

if (rightText.contains("RIGHT")) {

System.***out***.println("Right frame has the text 'RIGHT'.");

} else {

System.***out***.println("Right frame does not have the text 'RIGHT'.");

}

*driver*.switchTo().parentFrame();

*driver*.switchTo().defaultContent();

*driver*.switchTo().frame("frame-bottom");

String bottomText = *driver*.getPageSource();

if (bottomText.contains("BOTTOM")) {

System.***out***.println("Bottom frame has the text 'BOTTOM'.");

} else {

System.***out***.println("Bottom frame does not have the text 'BOTTOM'.");

}

*driver*.switchTo().parentFrame();

}

public static void waitForTime(int time) {

try {

Thread.*sleep*(time);

} catch (InterruptedException e) {

e.printStackTrace();

}

}

}

Output:

No of frames present in top frame:: 3

Left frame has the text 'LEFT'.

Middle frame has the text 'MIDDLE'.

Right frame has the text 'RIGHT'.

Bottom frame has the text 'BOTTOM'.