Source Code

1.Base Test: -

package com.redbus.base;

import java.io.File;

import java.io.IOException;

import java.lang.reflect.Method;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.chrome.ChromeDriver;

import org.openqa.selenium.firefox.FirefoxDriver;

import org.testng.Assert;

import org.testng.ITestContext;

import org.testng.ITestResult;

import org.testng.annotations.AfterMethod;

import org.testng.annotations.AfterSuite;

import org.testng.annotations.AfterTest;

import org.testng.annotations.BeforeSuite;

import org.testng.annotations.BeforeTest;

import java.awt.Desktop;

import com.aventstack.extentreports.ExtentReports;

import com.aventstack.extentreports.ExtentTest;

import com.aventstack.extentreports.reporter.ExtentSparkReporter;

public class BaseTest {

public static WebDriver driver;

public static ExtentReports extentreports;

public static ExtentTest extentTest;

public static void openBrowser(String browser) {

if (browser.equals("Chrome")) {

driver = new ChromeDriver();

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

Assert.assertTrue(true);

}

else if (browser.equals("Firefox")) {

driver = new FirefoxDriver();

Assert.assertTrue(true);

}

else {

System.out.println("Please Choose Chrome or Firefox Browser");

Assert.assertTrue(false);

}

}

@AfterTest

public static void closeBrowser() {

System.out.println("Browser closed");

driver.quit();

}

@BeforeTest

public void getnameMethod(ITestContext context)

{

extentTest= extentreports.createTest(context.getName());

}

@BeforeSuite // this method will be excuted before suite begins its execution

public void InitalizeExtentReport()

{

ExtentSparkReporter sparkreporter = new ExtentSparkReporter("report.html");

extentreports = new ExtentReports();

extentreports.attachReporter(sparkreporter);

// on the report display more information about OS, browser, java etc

extentreports.setSystemInfo("OS", System.getProperty("os.name"));

extentreports.setSystemInfo("JAVA", System.getProperty("java.version"));

}

@AfterSuite

public void generateReports() throws IOException

{

extentreports.flush();

Desktop.getDesktop().browse(new File("report.html").toURI());

}

@AfterMethod

public void generateTestStatus(Method m,ITestResult result)

{

if(result.getStatus() == ITestResult.FAILURE )

{

System.out.println("Capture Screenshot");

extentTest.fail(result.getThrowable());

}

else if (result.getStatus() == ITestResult.SUCCESS) {

extentTest.pass(m.getName() + " is passed");

}

}

}

2.Home Page: -

package com.redbus.pages;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.WebElement;

import org.openqa.selenium.support.FindBy;

import org.openqa.selenium.support.PageFactory;

public class HomePage {

// Create PageFactory

// this code will find the element using xpath locator and store it in webelement object myaccount

@FindBy(xpath = "//input[@id='src']")

WebElement fromLocation;

@FindBy(xpath = "//input[@id='dest']")

WebElement destination;

@FindBy(xpath="//div[@id='onwardCal']")

WebElement inputDate;

@FindBy(xpath="//span[text()='13']")

WebElement selectDate;

@FindBy(xpath="//button[@id='search\_button']")

WebElement search;

// create constructor of current class to initaiate the elements

// every page class will ahve a constructor initiating the elements

public HomePage(WebDriver driver) {

PageFactory.initElements(driver, this);

// driver can now work with your elements and methods of this class

}

// We will now write action methods, that will perform action on these elements

//but these methods will not be executed here.

public void gettitle(WebDriver driver)

{

String title = driver.getTitle();

System.out.println(title);

}

public void fromLocation(String locationName) throws InterruptedException {

fromLocation.sendKeys(locationName);

Thread.sleep(1000);

}

public void toLocation(String destName) throws InterruptedException {

destination.sendKeys(destName);

Thread.sleep(1000);

}

public void selectDate() throws InterruptedException {

inputDate.click();

Thread.sleep(1000);

selectDate.click();

}

public void search() {

search.click();

}

}

3.HomePage Test: -

package com.redbus.testpages;

import org.testng.annotations.BeforeTest;

import org.testng.annotations.Test;

import com.redbus.base.BaseTest;

import com.redbus.pages.HomePage;

public class HomePageTest extends BaseTest {

@BeforeTest

public void test() {

openBrowser("Firefox");

}

@Test

public void Home\_Page\_Test() throws InterruptedException {

HomePage homepg=new HomePage(driver);

driver.get("https://www.redbus.in/");

Thread.sleep(1000);

homepg.gettitle(driver);

homepg.fromLocation("Pune");

homepg.toLocation("Bangalore");

homepg.selectDate();

homepg.search();

}

}

4.POM XML: -

<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 https://maven.apache.org/xsd/maven-4.0.0.xsd">

<modelVersion>4.0.0</modelVersion>

<groupId>Selenium-Jenkins</groupId>

<artifactId>Selenium-Jenkins</artifactId>

<version>0.0.1-SNAPSHOT</version>

<dependencies>

<!-- https://mvnrepository.com/artifact/org.seleniumhq.selenium/selenium-java -->

<dependency>

<groupId>org.seleniumhq.selenium</groupId>

<artifactId>selenium-java</artifactId>

<version>4.11.0</version>

</dependency>

<!-- https://mvnrepository.com/artifact/io.github.bonigarcia/webdrivermanager -->

<dependency>

<groupId>io.github.bonigarcia</groupId>

<artifactId>webdrivermanager</artifactId>

<version>5.4.0</version>

</dependency>

<!-- https://mvnrepository.com/artifact/org.testng/testng -->

<dependency>

<groupId>org.testng</groupId>

<artifactId>testng</artifactId>

<version>7.7.0</version>

<scope>test</scope>

</dependency>

<!-- https://mvnrepository.com/artifact/com.aventstack/extentreports -->

<dependency>

<groupId>com.aventstack</groupId>

<artifactId>extentreports</artifactId>

<version>5.0.9</version>

</dependency>

</dependencies>

<build>

<pluginManagement><!-- lock down plugins versions to avoid using Maven defaults (may be moved to parent pom) -->

<plugins>

<!-- clean lifecycle, see https://maven.apache.org/ref/current/maven-core/lifecycles.html#clean\_Lifecycle -->

<plugin>

<artifactId>maven-clean-plugin</artifactId>

<version>3.1.0</version>

</plugin>

<!-- default lifecycle, jar packaging: see https://maven.apache.org/ref/current/maven-core/default-bindings.html#Plugin\_bindings\_for\_jar\_packaging -->

<plugin>

<artifactId>maven-resources-plugin</artifactId>

<version>3.0.2</version>

</plugin>

<plugin>

<artifactId>maven-compiler-plugin</artifactId>

<version>3.8.0</version>

</plugin>

<plugin>

<artifactId>maven-surefire-plugin</artifactId>

<version>2.22.1</version>

<configuration>

<suiteXmlFiles>

<!-- <suiteXmlFile>src/main/resources/TestRunner3.xml</suiteXmlFile> -->

<suiteXmlFile>src/test/resources/testng.xml</suiteXmlFile>

</suiteXmlFiles>

</configuration>

</plugin>

<plugin>

<artifactId>maven-jar-plugin</artifactId>

<version>3.0.2</version>

</plugin>

<plugin>

<artifactId>maven-install-plugin</artifactId>

<version>2.5.2</version>

</plugin>

<plugin>

<artifactId>maven-deploy-plugin</artifactId>

<version>2.8.2</version>

</plugin>

<!-- site lifecycle, see https://maven.apache.org/ref/current/maven-core/lifecycles.html#site\_Lifecycle -->

<plugin>

<artifactId>maven-site-plugin</artifactId>

<version>3.7.1</version>

</plugin>

<plugin>

<artifactId>maven-project-info-reports-plugin</artifactId>

<version>3.0.0</version>

</plugin>

</plugins>

</pluginManagement>

</build>

</project>

5.TestNg XML: -

<suite name="test">

<test name="TestHomePage">

<classes>

<class name="com.redbus.testpages.HomePageTest"></class>

<class name="com.redbus.base.BaseTest"></class>

</classes>

</test>

</suite>