TASK1:

CODE:

package day9;

import org.testng.annotations.Test;

import org.testng.Assert;

import org.testng.annotations.DataProvider;

public class Task1 {

int a,b,c;

@Test(dataProvider = "dp",priority = 1)

public void add(Integer a, Integer b) {

c = a+b;

System.out.println("Sum = "+c);

Assert.assertEquals(c,15);

}

@Test(dataProvider = "dp",priority = 2)

public void subtract(Integer a, Integer b) {

c = a-b;

System.out.println("Difference = "+c);

Assert.assertEquals(c,5);

}

@Test(dataProvider = "dp",priority = 3)

public void multiply(Integer a, Integer b) {

c = a\*b;

System.out.println("Multiplication = "+c);

Assert.assertEquals(c,50);

}

@Test(dataProvider = "dp",priority = 4)

public void divide(Integer a, Integer b) {

c = a/b;

System.out.println("Division = "+c);

Assert.assertEquals(c,2);

}

@DataProvider

public Object[][] dp() {

return new Object[][] {

new Object[] { 10, 5 },

};

}

}

OUTPUT:



TASK2&3:

CODE:

package day9;

import org.openqa.selenium.firefox.\*;

import java.util.concurrent.TimeUnit;

import org.apache.hc.core5.util.Timeout;

import org.openqa.selenium.By;

import org.openqa.selenium.WebDriver;

import org.testng.Assert;

import org.testng.annotations.BeforeTest;

import org.testng.annotations.Test;

import io.github.bonigarcia.wdm.WebDriverManager;

public class Task2\_3 {

WebDriver driver;

@Test(groups = {"SmokeTest"})

public void testcase1() throws InterruptedException {

WebDriverManager.firefoxdriver().setup();

FirefoxOptions co = new FirefoxOptions();

driver = new FirefoxDriver(co);

driver.get("https://www.godaddy.com/en-in");

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

String s1 = driver.getTitle();

Assert.assertEquals(s1,"Domain Names, Websites, Hosting & Online Marketing Tools - GoDaddy IN");

String url1 = driver.getCurrentUrl();

Assert.assertEquals(url1,"https://www.godaddy.com/en-in");

Thread.sleep(2000);

// driver.close();

}

@Test(groups = {"Regression TestCase"})

public void testcase2() throws InterruptedException

{

Thread.sleep(2000);

WebDriverManager.firefoxdriver().setup();

FirefoxOptions co = new FirefoxOptions();

driver = new FirefoxDriver(co);

driver.get("https://www.godaddy.com/en-in");

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

Thread.sleep(2000);

driver.manage().timeouts().implicitlyWait(2,TimeUnit.SECONDS);

Thread.sleep(2000);

driver.findElement(By.xpath("//\*[@id=\"id-631b049a-e9c0-4d24-8710-c504745206dd\"]/div[2]/div[1]/ul/li[1]/button")).click();

Thread.sleep(2000);

driver.findElement(By.xpath("//\*[@id=\"id-631b049a-e9c0-4d24-8710-c504745206dd\"]/div[2]/div[1]/ul/li[1]/div/div[2]/div[1]/ul/li[2]/a")).click();

// driver.close();

}

@Test(groups = {"Regression TestCase"})

public void testcase3() throws InterruptedException

{

Thread.sleep(2000);

WebDriverManager.firefoxdriver().setup();

FirefoxOptions co = new FirefoxOptions();

driver = new FirefoxDriver(co);

driver.get("https://www.godaddy.com/en-in");

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

Thread.sleep(2000);

driver.manage().timeouts().implicitlyWait(2,TimeUnit.SECONDS);

Thread.sleep(2000);

driver.findElement(By.xpath("//\*[@id=\"id-631b049a-e9c0-4d24-8710-c504745206dd\"]/div[2]/div[1]/ul/li[1]/button")).click();

Thread.sleep(2000);

driver.findElement(By.linkText("Domain Name Search")).click();

Thread.sleep(2000);

String s =driver.getTitle();

System.out.println(s);

Assert.assertEquals(s,"GoDaddy Domain Search - Buy and Register Available Domain Names");

}

}

OUTPUT:



TASK4:

CODE:

package day9;

import org.openqa.selenium.By;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.edge.\*;

import org.openqa.selenium.firefox.\*;

import org.testng.Assert;

import org.testng.annotations.AfterMethod;

import org.testng.annotations.BeforeTest;

import org.testng.annotations.Parameters;

import org.testng.annotations.Test;

import io.github.bonigarcia.wdm.WebDriverManager;

public class Task4 {

WebDriver driver;

@Parameters({"browser"})

@BeforeTest

public void open(String ed) {

if(ed.equals("edge"))

{

WebDriverManager.edgedriver().setup();

EdgeOptions co = new EdgeOptions();

co.addArguments("--remote-allow-origins=\*");

driver = new EdgeDriver(co);

driver.get("https://opensource-demo.orangehrmlive.com/web/index.php/auth/login");

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

}

else if(ed.equals("FireFox"))

{

WebDriverManager.firefoxdriver().setup();

FirefoxOptions co = new FirefoxOptions();

co.addArguments("--remote-allow-origins=\*");

driver = new FirefoxDriver(co);

driver.get("https://opensource-demo.orangehrmlive.com/web/index.php/auth/login");

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

}

}

@Test

public void pageTest() throws InterruptedException

{

Thread.sleep(3000);

driver.findElement(By.name("username")).sendKeys("Admin");

Thread.sleep(1000);

driver.findElement(By.name("password")).sendKeys("admin123");

Thread.sleep(1000);

String s1 = driver.getCurrentUrl();

driver.findElement(By.xpath("//\*[@id=\"app\"]/div[1]/div/div[1]/div/div[2]/div[2]/form/div[3]/button")).click();

String s2 = driver.getCurrentUrl();

Assert.assertNotEquals(s1,s2);

}

@AfterMethod

public void afterThat() throws InterruptedException

{

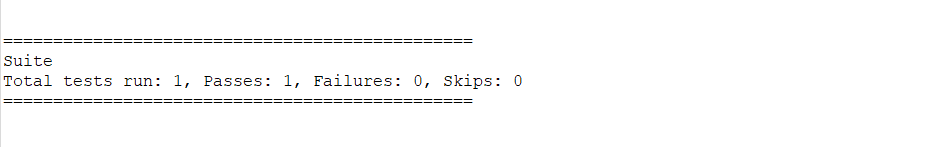
Thread.sleep(3000);

driver.close();

}

}

OUTPUT:



XML:

<?xml version="1.0" encoding="UTF-8"?>

<suite parallel="false" name="Suite">

<test name="Test">

<parameter name ="browser" value= "FireFox"/>

<classes>

<class name="day9.Task4"/>

</classes>

</test> <!-- Test -->

</suite> <!-- Suite -->

TASK5:

CODE:

package day9;

import org.openqa.selenium.By;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.edge.\*;

import org.testng.Assert;

import org.testng.annotations.AfterMethod;

import org.testng.annotations.BeforeMethod;

import org.testng.annotations.Test;

import io.github.bonigarcia.wdm.WebDriverManager;

public class Task5 {

WebDriver driver;

@BeforeMethod

public void openPage() {

WebDriverManager.edgedriver().setup();

EdgeOptions co = new EdgeOptions();

co.addArguments("--remote-allow-origins=\*");

driver = new EdgeDriver(co);

driver.get("https://opensource-demo.orangehrmlive.com/web/index.php/auth/login");

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

}

@Test

public void pageTest() throws InterruptedException

{

Thread.sleep(3000);

driver.findElement(By.name("username")).sendKeys("Suvitha");

Thread.sleep(1000);

driver.findElement(By.name("password")).sendKeys("12345");

Thread.sleep(1000);

String s1 = driver.getCurrentUrl();

driver.findElement(By.xpath("//\*[@id=\"app\"]/div[1]/div/div[1]/div/div[2]/div[2]/form/div[3]/button")).click();

String s2 = driver.getCurrentUrl();

Assert.assertNotEquals(s1,s2);

}

@AfterMethod

public void afterThat() throws InterruptedException

{

Thread.sleep(3000);

driver.close();

}

}

LISTENER:

package day9;

import org.testng.ITestContext;

import org.testng.ITestListener;

import org.testng.ITestResult;

public class Task5Listener implements ITestListener{

public void onTestStart(ITestResult result) {

// TODO Auto-generated method stub

System.out.println("on Test Start "+result.getName());

}

public void onTestSuccess(ITestResult result) {

// TODO Auto-generated method stub

System.out.println("on Test Success "+result.getName());

}

public void onTestFailure(ITestResult result) {

// TODO Auto-generated method stub

System.out.println("on Test Failure "+result.getName());

}

public void onTestSkipped(ITestResult result) {

// TODO Auto-generated method stub

System.out.println("on Test Skipped "+result.getName());

}

public void onTestFailedButWithinSuccessPercentage(ITestResult result) {

// TODO Auto-generated method stub

System.out.println("on Test Failed But Within Success Percentage "+result.getName());

}

public void onTestFailedWithTimeout(ITestResult result) {

// TODO Auto-generated method stub

System.out.println("on Test Failed With Timeout "+result.getName());

}

public void onStart(ITestContext context) {

// TODO Auto-generated method stub

System.out.println("on Start "+context.getName());

}

public void onFinish(ITestContext context) {

// TODO Auto-generated method stub

System.out.println("on Finish "+context.getName());

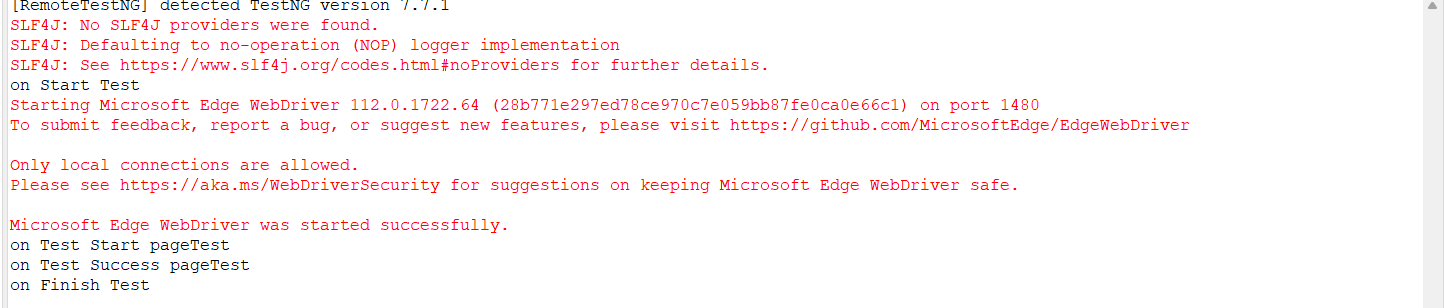
}

public static void main(String[] args) {

}

}

OUTPUT:



XML:

<?xml version="1.0" encoding="UTF-8"?>

<suite parallel="false" name="Suite">

<test name="Test">

<listeners>

<listener class-name = "day9.Task5Listener"/>

</listeners>

<classes>

<class name="day9.Task5"/>

</classes>

</test> <!-- Test -->

</suite> <!-- Suite -->