AUTOMATION SELENIUM IDE

LAB EXERCISE

DAY-10

package com.Test.test;

import org.testng.annotations.Test;

import java.util.concurrent.TimeUnit;

import org.openqa.selenium.By;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.chrome.ChromeDriver;

import org.openqa.selenium.chrome.ChromeOptions;

import org.testng.Assert;

import org.testng.ITestResult;

import org.testng.annotations.AfterMethod;

import org.testng.annotations.AfterTest;

import org.testng.annotations.BeforeClass;

import org.testng.annotations.Test;

import com.aventstack.extentreports.ExtentReports;

import com.aventstack.extentreports.ExtentTest;

import com.aventstack.extentreports.Status;

import com.aventstack.extentreports.reporter.ExtentHtmlReporter;

import com.aventstack.extentreports.reporter.configuration.ChartLocation;

import com.aventstack.extentreports.reporter.configuration.Theme;

import io.github.bonigarcia.wdm.WebDriverManager;

public class D10C1 {

static ExtentTest test;

static ExtentReports extent;

ExtentHtmlReporter htmlReporter;

@BeforeClass

public void startReport() {

//ExtentSparkReporter htmlReporter = new ExtentSparkReporter("extentReport.html");

//htmlReporter = new ExtentHtmlReporter(System.getProperty("user.dir")+"/test-output/extentReport.html");

htmlReporter = new ExtentHtmlReporter("E:\\extr2.html");

extent = new ExtentReports();

extent.attachReporter(htmlReporter);

//configuration items to change the look and feel

//add content, manage tests etc

htmlReporter.config().setChartVisibilityOnOpen(true);

htmlReporter.config().setDocumentTitle("Simple Automation Report");

htmlReporter.config().setReportName("Test Report");

htmlReporter.config().setTestViewChartLocation(ChartLocation.TOP);

htmlReporter.config().setTheme(Theme.STANDARD);

// htmlReporter.config().setTimeStampFormat("EEEE, MMMM dd, yyyy, hh:mm a '('zzz')'");

}

@Test

public void f() {

WebDriverManager.chromedriver().setup();

WebDriver driver;

ChromeOptions co = new ChromeOptions();

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

driver=new ChromeDriver(co);

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

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

String actualTitle="Domain Names, Websites, Hosting & Online Marketing Tools - GoDaddy IN";

String extractedTitle=driver.getTitle();

Assert.assertEquals(actualTitle, extractedTitle);

String actualURL="https://www.godaddy.com/en-in";

String extractedURL=driver.getCurrentUrl();

Assert.assertEquals(actualURL, extractedURL);

System.out.println("Smoke test");

driver.quit();

test = extent.createTest("Test Case 1", "The test case 1 has passed");

}

@Test

public void f1() throws Exception {

WebDriverManager.chromedriver().setup();

WebDriver driver;

ChromeOptions co = new ChromeOptions();

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

driver=new ChromeDriver(co);

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

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

driver.navigate().refresh();

//driver.manage().timeouts().implicitlyWait(5, TimeUnit.SECONDS);

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

Thread.sleep(10);

driver.findElement(By.xpath("//\*[@id=\"id-e6aaf13d-272e-44df-903a-e07e539d57c7\"]/span[1]")).click();

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

System.out.println(driver.getTitle());

System.out.println("Regressiontest");

test = extent.createTest("Test Case 2", "The test case 2 has passed");

//driver.quit();

}

@AfterMethod

public void getResult(ITestResult result) {

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

test.log(Status.FAIL,result.getThrowable());

}

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

test.log(Status.PASS, result.getTestName());

}

else {

test.log(Status.SKIP, result.getTestName());

}

}

@AfterTest

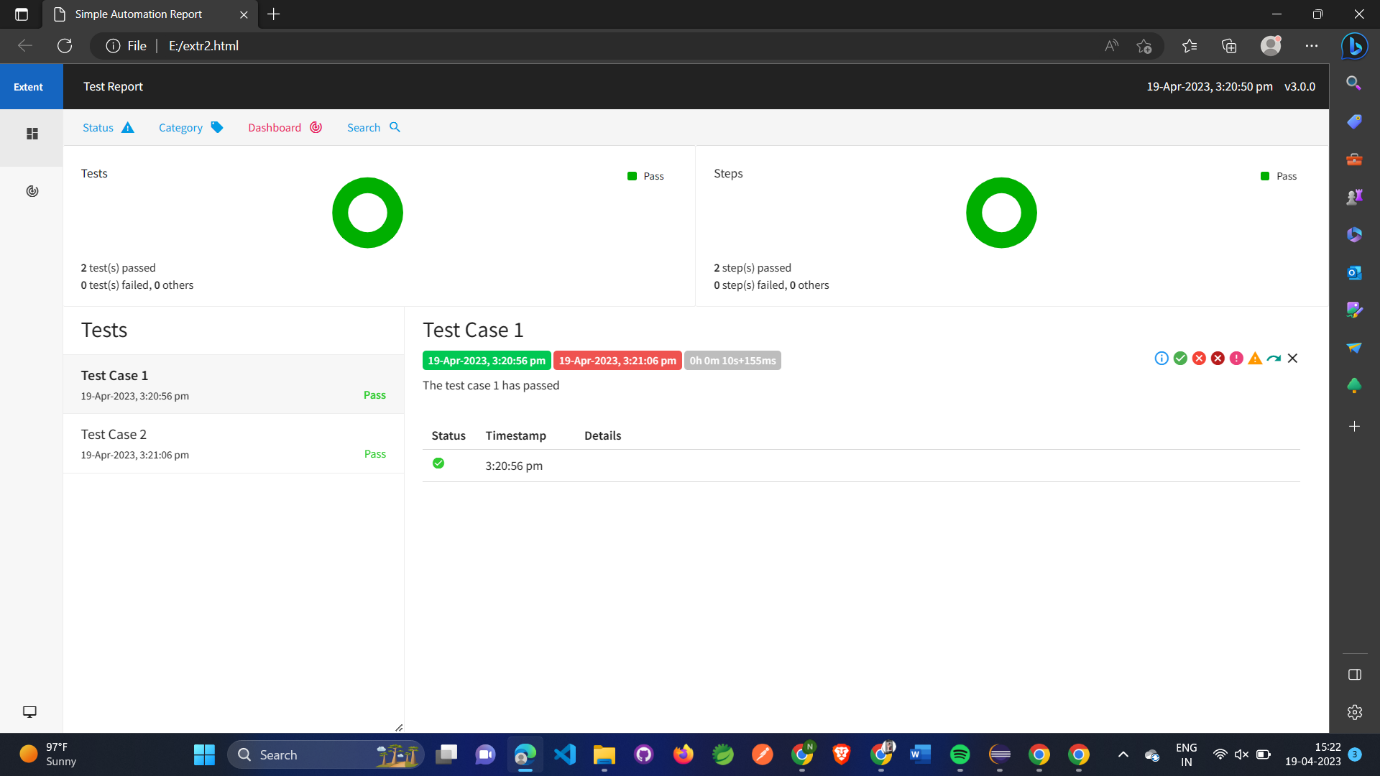
public void tearDown() {

//to write or update test information to reporter

extent.flush();

}

}



package com.Test.test;

import org.testng.annotations.Test;

import java.util.concurrent.TimeUnit;

import org.openqa.selenium.By;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.chrome.ChromeDriver;

import org.openqa.selenium.chrome.ChromeOptions;

import org.testng.Assert;

import org.testng.ITestResult;

import org.testng.annotations.AfterMethod;

import org.testng.annotations.AfterTest;

import org.testng.annotations.BeforeClass;

import org.testng.annotations.Test;

import com.aventstack.extentreports.ExtentReports;

import com.aventstack.extentreports.ExtentTest;

import com.aventstack.extentreports.Status;

import com.aventstack.extentreports.reporter.ExtentHtmlReporter;

import com.aventstack.extentreports.reporter.configuration.ChartLocation;

import com.aventstack.extentreports.reporter.configuration.Theme;

import io.github.bonigarcia.wdm.WebDriverManager;

public class D10C1 {

static ExtentTest test;

static ExtentReports extent;

ExtentHtmlReporter htmlReporter;

@BeforeClass

public void startReport() {

//ExtentSparkReporter htmlReporter = new ExtentSparkReporter("extentReport.html");

//htmlReporter = new ExtentHtmlReporter(System.getProperty("user.dir")+"/test-output/extentReport.html");

htmlReporter = new ExtentHtmlReporter("E:\\extr2.html");

extent = new ExtentReports();

extent.attachReporter(htmlReporter);

//configuration items to change the look and feel

//add content, manage tests etc

htmlReporter.config().setChartVisibilityOnOpen(true);

htmlReporter.config().setDocumentTitle("Simple Automation Report");

htmlReporter.config().setReportName("Test Report");

htmlReporter.config().setTestViewChartLocation(ChartLocation.TOP);

htmlReporter.config().setTheme(Theme.STANDARD);

// htmlReporter.config().setTimeStampFormat("EEEE, MMMM dd, yyyy, hh:mm a '('zzz')'");

}

@Test

public void f() {

WebDriverManager.chromedriver().setup();

WebDriver driver;

ChromeOptions co = new ChromeOptions();

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

driver=new ChromeDriver(co);

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

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

String actualTitle="Domain Names, Websites, Hosting & Online Marketing Tools - GoDaddy IN";

String extractedTitle=driver.getTitle();

Assert.assertEquals(actualTitle, extractedTitle);

String actualURL="https://www.godaddy.com/en-in";

String extractedURL=driver.getCurrentUrl();

Assert.assertEquals(actualURL, extractedURL);

System.out.println("Smoke test");

driver.quit();

test = extent.createTest("Test Case 1", "The test case 1 has passed");

}

@Test

public void f1() throws Exception {

WebDriverManager.chromedriver().setup();

WebDriver driver;

ChromeOptions co = new ChromeOptions();

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

driver=new ChromeDriver(co);

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

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

driver.navigate().refresh();

//driver.manage().timeouts().implicitlyWait(5, TimeUnit.SECONDS);

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

Thread.sleep(10);

driver.findElement(By.xpath("//\*[@id=\"id-e6aaf13d-272e-44df-903a-e07e539d57c7\"]/span[1]")).click();

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

System.out.println(driver.getTitle());

System.out.println("Regressiontest");

test = extent.createTest("Test Case 2", "The test case 2 has passed");

//driver.quit();

}

@Test

public void f2() {

ChromeOptions co=new ChromeOptions();

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

WebDriverManager.chromedriver().setup();

WebDriver d=new ChromeDriver(co);

d=new ChromeDriver(co);

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

d.manage().window().maximize();

d.manage().timeouts().implicitlyWait

(10,TimeUnit.SECONDS);

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

d.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();

String et="GoDaddy Domain Search - Buy and Register Available Domain Names";

String ct=d.getTitle();

System.out.println(ct);

Assert.assertEquals(et,ct);

System.out.println("Regressiontest");

test = extent.createTest("Test Case 3", "The test case 3 has passed");

}

@AfterMethod

public void getResult(ITestResult result) {

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

test.log(Status.FAIL,result.getThrowable());

}

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

test.log(Status.PASS, result.getTestName());

}

else {

test.log(Status.SKIP, result.getTestName());

}

}

@AfterTest

public void tearDown() {

//to write or update test information to reporter

extent.flush();

}

}

