105. Handling HTTPS certifications in Automated browsers

Expired.badssl.com

First create object from the chrome driver ( webdriver driver = new chromedriver() )

Driver.get(“url”);

Print title of that page

Sometimes they will show this certification at that time, guys, see you are getting the wrong title.

So in that case, there is one simple class in chrome called chrome options.

this chrome options class help you to set the behavior for your Chrome browser.

 first, I will create object for this Chrome actions class.

So with this object, you can access the methods present in these Chrome actions class.

So you are internally setting a chrome behavior to accept the certificates and proceed to the actual website.

This options object have the behavior of how it should behave so that behavior options object.

You have to pass as an argument to your chrome driver class.

Then only this chrome driver will get the knowledge about how it has to invoke the browser.

Now run your test.

It should work like it so that it should bypass that SSL and it will go to the actual webpage.

so that's how you can bypass this in security certification site SSL certifications.

ChromeOptions options = **new** ChromeOptions();

options.setAcceptInsecureCerts(**true**);

106. Explore Chrome options to set proxies,plugins & paths on Chrome browser

Now let's say you want to open your automation browser by default.

Some extensions.

Chromedriver.chromium.org\capabilities

107. Code download

**import** java.util.HashMap;

**import** java.util.Map;

**import** org.openqa.selenium.Proxy;

**import** org.openqa.selenium.WebDriver;

**import** org.openqa.selenium.chrome.ChromeDriver;

**import** org.openqa.selenium.chrome.ChromeOptions;

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

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

**public** **class** SSLCheck {

**public** **static** **void** main(String[] args) {

// **TODO** Auto-generated method stub

ChromeOptions options = **new** ChromeOptions();

Proxy proxy = **new** Proxy();

proxy.setHttpProxy("ipaddress:4444");

options.setCapability("proxy", proxy);

Map<String, Object> prefs = **new** HashMap<String, Object>();

prefs.put("download.default\_directory", "/directory/path");

options.setExperimentalOption("prefs", prefs);

// FirefoxOptions options1 = new FirefoxOptions();

// options1.setAcceptInsecureCerts(true);

// EdgeOptions options2 = new EdgeOptions();

options.setAcceptInsecureCerts(**true**);

System.setProperty("webdriver.chrome.driver", "/Users/rahulshetty/Documents/chromedriver");

WebDriver driver = **new** ChromeDriver(options);

driver.get("https://expired.badssl.com/");

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

}

}

108. Maximizing window and deleting cookies

how to maximize the browser and how to delete the cookies before you start testing.

If you want to execute your tests in maximized browser, you can do that with selenium.

Driver.manage().window().maximize();

if you want to delete all the cookies before you start the execution.

Most probably when you run your tests, selenium will invoke in a fresh instance by deleting all the cookies.

But let's say you have done for 1 minutes or 2 minutes of your execution on the browser.

So after two minutes you have a test like you need to delete all the cookies in the browser and make

sure that elements are not pre-populating in the website.

There could be some scenario where you have to come across deleting the cookies before proceeding.

Driver.manage.deleteAllCookies();

if you don't want to delete all the cookies and if you want to delete only one cookie and

start validating some scenarios.

Driver.manage.deletecookiename(“mentione that specific cookie name “);

Add the cookie.

You can even add the cookies at runtime from your browser.

Interview que : So they want to check the scenario where they want to delete the session cookie and make sure that secure login gets logout and he has to navigate to the home page. So how do you automate that?

So in this case you have to write this step to delete the cookie and click on any link and then validate if you are in the login URL.

So we know now how to maximize the browser, how to delete the cookies, and also how to handle Https certifications and how to handle SSL certifications.

109. How to take Screenshots in Selenium

 let's see on how to delete the screenshots and also how to kill the process which

is running in the background.

So if you want to take a screenshot you have to convert your webdriver object into screenshot object.

So this object know how to automate your browser, but now you want to change this object behavior to capturing screenshot.

File utils jar download - [Commons IO – Commons IO Overview (apache.org)](https://commons.apache.org/proper/commons-io/)

driver.get("http://google.com");

File src= ((TakesScreenshot)driver).getScreenshotAs(OutputType.FILE);

FileUtils.copyFile(src,new File("C:\\Users\\rahul\\screenshot.png"));

110. Strategy to automate the broken Links with Selenium

111. Open Connection method to identify status codes of the links

Interview que : how to handle broken links in your application using selenium.

Real time - if you work in any e-commerce application, there will be at least 70 to 100 links in just your homepage.

how do you check whether all the links are perfectly working and now getting to the right pages?

So from your developer tools.

Select a network tap in the console and select xhr

f you see any status code greater than 400, like four or one, four, three, 500.

So that means those links are broken. – this one way to guess

we will try to call that you all and see what is the status code.

So here there are some Java methods which will help you to call those url and will gets you the status quotes

So using selenium, first we will land on the page where you have to verify broken links and you will scan all the links using selenium.

And for each link, you will call attribute method and you will pull up this url

so first to step step one is to get all the url’s tied up to the links using selenium.

Once you get url’s in your hand, then you will use Jawa methods to call those url and get status codes

if status code is >400, then that url is is not working, which you can say eventually the link which is tied up to that url is broken.

so there is a method called open connection, so this method comes from url class, OK, that is already a predefined class called url in Java, which exposes this open

connection method.

To call any method you can call that method using object of that class.

Once you create object of the class, then object. method.

This is how you can call any method belongs to that class.

SO openconnection method is exposed by a class called url I will create object of this class. a new operator and class name like this, that object will be created.

( new URL(). open connection(); )

/broken URL

         //Step 1 - IS to get all urls tied up to the links using Selenium

         //  Java methods will call URL's and gets you the status code

         //if status code >400 then that url is not working-> link which tied to url is broken

         //a[href\*="soapui"]'

driver.get("https://rahulshettyacademy.com/AutomationPractice/");

string url = drivr.findElement(By.cssSelector(“a[href\*=’brokenlink’]”)).getAttribute(“href”);

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

          conn.setRequestMethod("HEAD");

          conn.connect();

          int respCode = conn.getResponseCode();

          System.out.println(respCode);

Without spending much time off clicking that link and checking if that url is loading

So we actually got that you are in and we use some Java method open connection.

We sent a request called We get the response as it is 200. You can say that this link is not broken.

112. Iterate over all links in the page to validate broken Links mechanism

So now our goal is to scan each and every link of this footer section

I want to store all these items into the list and this list is nothing but a list of elements.

List<WebElements> links = driver.findelements(By.cssselector(“li[class=’gf-li’] a”));

We put all the links in the links variable

Now let's iterate this list

For ( webElements link : links )

So this is enhanced of a loop.

So basically now out of all the Web elements present in this link, for every iteration, it will pull one Web element and we are calling that Web element as a link.

 if you have 100 links in your page and if you are asked to identify

if all links are working properly, you can write that code like this and live it and it will simply

scan all the hundred links.

And if any link is fault, it will stop, break your test and you can read the log to understand.

113. Importance of Soft Assertions in Selenium WebDriver

So if you want an session which should not stop your execution, then there is a concept called soft assertion intestNG.

In google – testing jar dependency

If you don't put at the ( a.assertAll()) then it will catch all failures, but it will not report.

So if it is not reporting, then what is the purpose of using it?

What is soft and hard assertions ?

114. Code download

package tests;

import java.io.IOException;

import java.net.HttpURLConnection;

import java.net.MalformedURLException;

import java.net.URL;

import java.net.URLConnection;

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;

import org.testng.Assert;

import org.testng.asserts.SoftAssert;

public class BrokenLinks {

public static void main(String[] args) throws MalformedURLException, IOException {

// TODO Auto-generated method stub

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

         WebDriver driver=new ChromeDriver();

         //broken URL

         //Step 1 - IS to get all urls tied up to the links using Selenium

         //  Java methods will call URL's and gets you the status code

         //if status code >400 then that url is not working-> link which tied to url is broken

         //a[href\*="soapui"]'

         driver.get("https://rahulshettyacademy.com/AutomationPractice/");

      List<WebElement> links=   driver.findElements(By.cssSelector("li[class='gf-li'] a"));

      SoftAssert a =new SoftAssert();

      for(WebElement link : links)

      {

          String url= link.getAttribute("href");

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

          conn.setRequestMethod("HEAD");

          conn.connect();

          int respCode = conn.getResponseCode();

          System.out.println(respCode);

          a.assertTrue(respCode<400, "The link with Text"+link.getText()+" is broken with code" +respCode);

      }

      a.assertAll();

}

private static Object getReturnCode(WebElement link) {

// TODO Auto-generated method stub

return null;

}

}

Question 1:

Name the class to customize the Chrome browser.

Ans : ChromeOptions

Question 2:

Identify the method to maximize the browser window.

Ans : driver.manage().window().maximize()

Question 3:

How to delete cookies in Selenium Web Driver?

Ans : 1. Driver.manage().deleteallcookies()

2. driver.manage().deletecookienamed(“sessionid”)

Question 4:

How to capture screenshots with Selenium Web driver?

File s= ((TakesScreenshot)driver).getScreenshotAs(OutputType.FILE);

FileUtils.copyFile(src,new File("C:\\Users\\rahul\\screenshot.png"));

Question 5:

How to identify broken links in Selenium Webdriver?

Get status code of links to decide if they are broken