**How To Find Broken Links Using Selenium WebDriver**

One of the key test case is to find broken links on a webpage. Due to existence of broken links, your website reputation gets damaged and there will be a negative impact on your business. It’s mandatory to find and fix all the broken links before release. If a link is not working, we face a message as 404 Page Not Found.

Let’s see some of the HTTP status codes.

200 – Valid Link

404 – Link not found

400 – Bad request

401 – Unauthorized

500 – Internal Error

Consider a test case to test all the links in the home page of “http://newtours.demoaut.com"

Below code fetches all the links of a given website (i.e., http://newtours.demoaut.com) using ***WebDriver***commands and reads the status of each *href* link with the help of ***HttpURLConnection*** class.

**Example:**

import java.io.IOException;  
import java.net.HttpURLConnection;  
import java.net.URL;  
import java.util.List;  
import java.util.concurrent.TimeUnit;  
import org.openqa.selenium.By;  
import org.openqa.selenium.WebDriver;  
import org.openqa.selenium.WebElement;  
import org.openqa.selenium.chrome.ChromeDriver;  
  
public class BrokenLinks {  
 public static void main(String[] args) throws InterruptedException, IOException {  
  
 // Instantiating FirefoxDriver  
 System.setProperty("webdriver.chrome.driver", "C://Drivers/chromedriver\_win32/chromedriver.exe");  
 WebDriver driver = new ChromeDriver();  
  
 // Maximize the browser  
 driver.manage().window().maximize();  
  
 // Implicit wait for 10 seconds  
 driver.manage().timeouts().implicitlyWait(10, TimeUnit.SECONDS);  
  
 // To launch pavantestingtools.com  
 driver.get("http://newtours.demoaut.com/");  
  
 // Wait for 5 seconds  
 Thread.sleep(5000);  
  
 // Used tagName method to collect the list of items with tagName "a"  
 // findElements - to find all the elements with in the current page. It  
 // returns a list of all webelements or an empty list if nothing matches  
 List links = driver.findElements(By.tagName("a"));  
  
 // To print the total number of links  
 System.out.println("Total links are " + links.size());  
  
 // used for loop to  
 for (int i = 0; i < links.size(); i++) {  
 WebElement element = links.get(i);  
 // By using "href" attribute, we could get the url of the requried  
 // link  
 String url = element.getAttribute("href");  
  
 URL link = new URL(url);  
 // Create a connection using URL object (i.e., link)  
 HttpURLConnection httpConn = (HttpURLConnection) link.openConnection();  
 // Set the timeout for 2 seconds  
 httpConn.setConnectTimeout(2000);  
 // connect using connect method  
 httpConn.connect();  
 // use getResponseCode() to get the response code.  
 if (httpConn.getResponseCode() >= 400) {  
 System.out.println(url + " - " + "is Broken Link");  
 } else {  
 System.out.println(url + " - " + "is valid Link");  
 }  
 }  
 }  
}