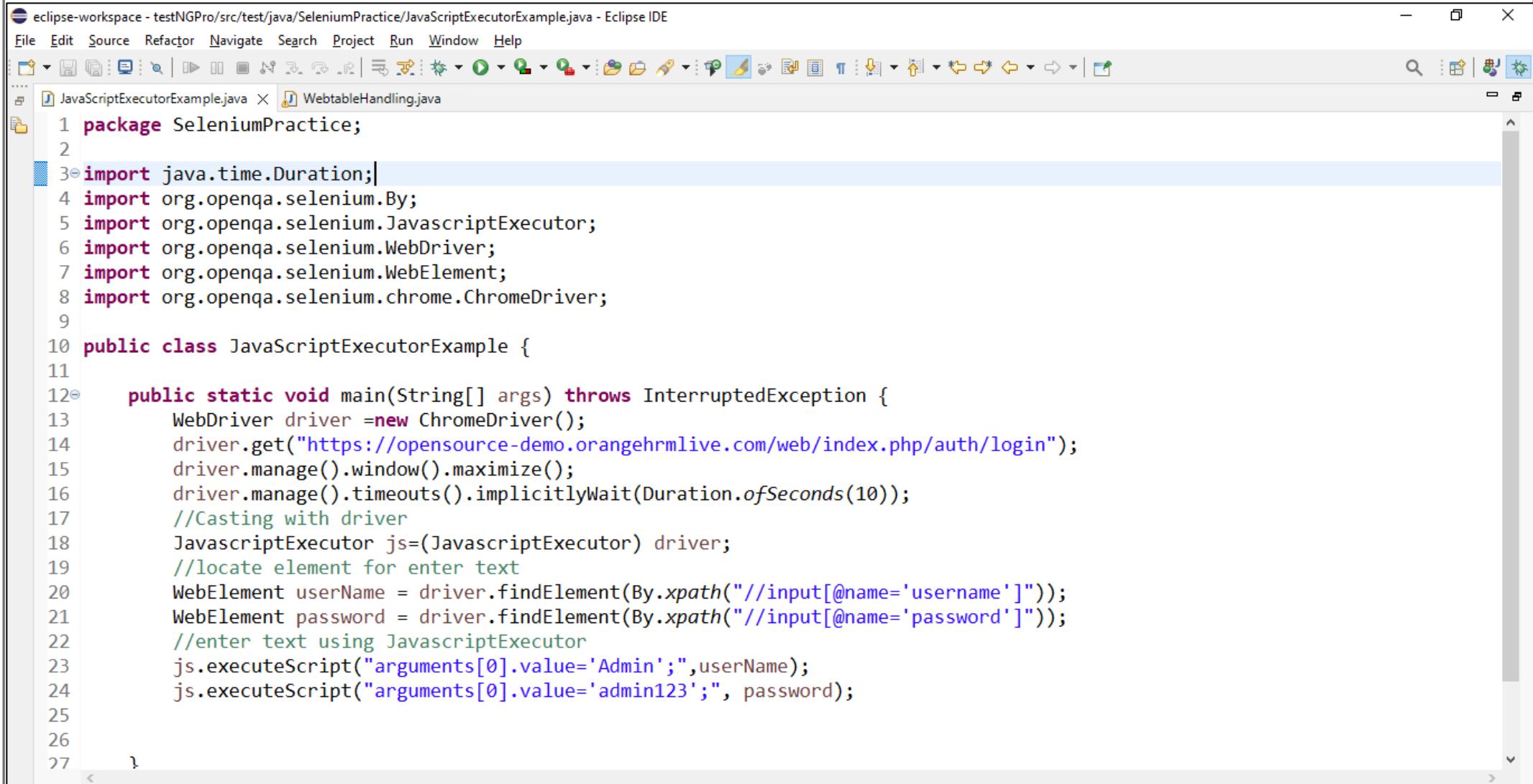


1. To Click on a Button

```
1 package SeleniumPractice;
2
3 import java.time.Duration;
4
5 import org.openqa.selenium.By;
6 import org.openqa.selenium.JavascriptExecutor;
7 import org.openqa.selenium.WebDriver;
8 import org.openqa.selenium.WebElement;
9 import org.openqa.selenium.chrome.ChromeDriver;
10
11 public class JavaScriptExecutorExample {
12
13 public static void main(String[] args) {
14     WebDriver driver =new ChromeDriver();
15     driver.get("https://opensource-demo.orangehrmlive.com/web/index.php/auth/login");
16     driver.manage().window().maximize();
17     driver.manage().timeouts().implicitlyWait(Duration.ofSeconds(10));
18     //Locate the target element for click
19     WebElement loginBtn = driver.findElement(By.xpath("//button[text()=' Login ']"));
20     //Up casting JavascriptExecutor with driver
21     JavascriptExecutor js=(JavascriptExecutor) driver;
22     //Click element by JavascriptExecutor
23     js.executeScript("arguments[0].click();", loginBtn);
24
25 }
26
27 }
```

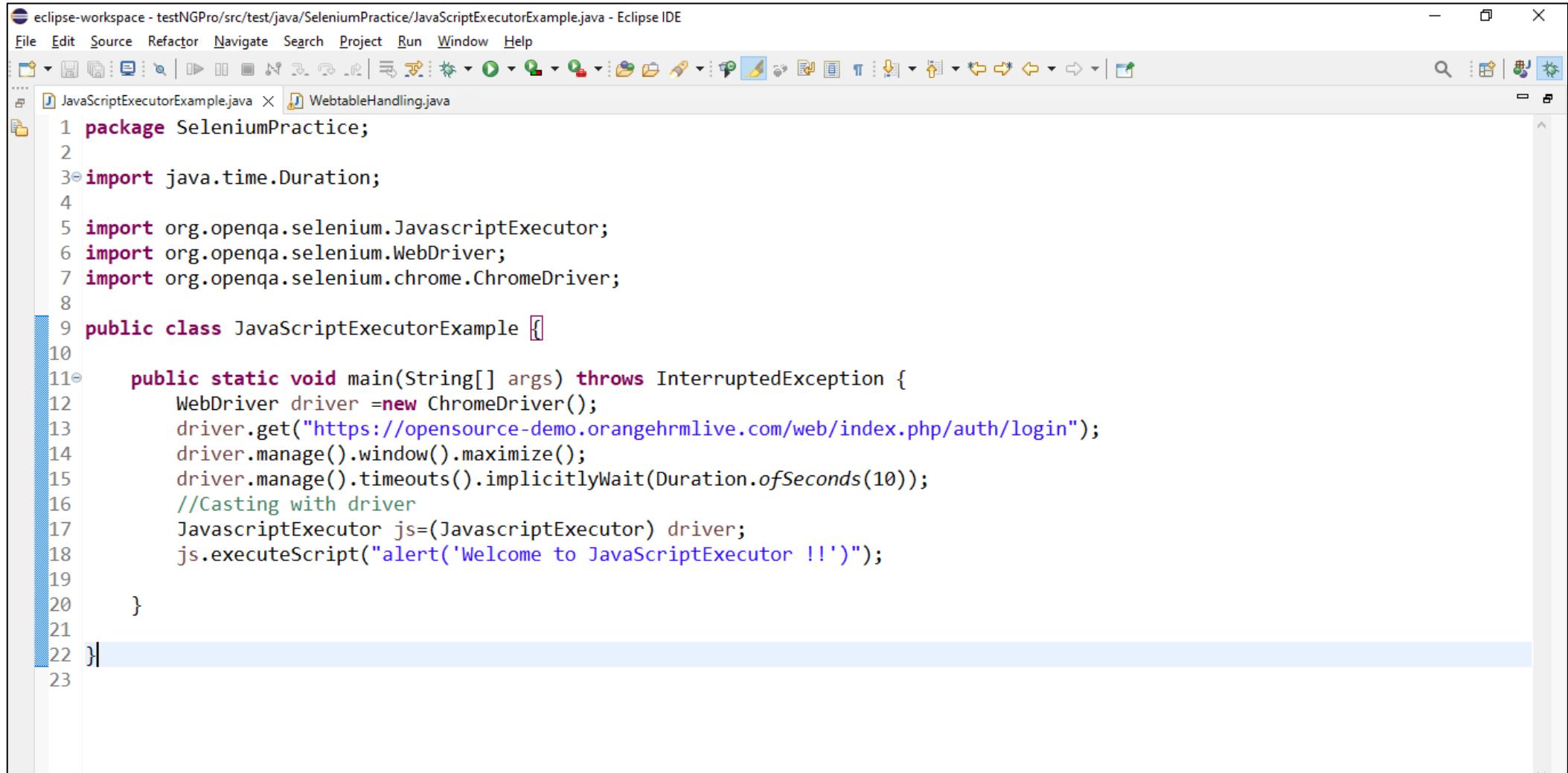
2. To Type Text in a Text Box without using sendKeys() method



The screenshot shows the Eclipse IDE interface with the title bar "eclipse-workspace - testNGPro/src/test/java/SeleniumPractice/JavaScriptExecutorExample.java - Eclipse IDE". The menu bar includes File, Edit, Source, Refactor, Navigate, Search, Project, Run, Window, and Help. The toolbar has various icons for file operations like Open, Save, Find, and Run. The left sidebar shows project files "JavaScriptExecutorExample.java X" and "WebtableHandling.java". The main editor area contains the following Java code:

```
1 package SeleniumPractice;
2
3 import java.time.Duration;
4 import org.openqa.selenium.By;
5 import org.openqa.selenium.JavascriptExecutor;
6 import org.openqa.selenium.WebDriver;
7 import org.openqa.selenium.WebElement;
8 import org.openqa.selenium.chrome.ChromeDriver;
9
10 public class JavaScriptExecutorExample {
11
12     public static void main(String[] args) throws InterruptedException {
13         WebDriver driver =new ChromeDriver();
14         driver.get("https://opensource-demo.orangehrmlive.com/web/index.php/auth/login");
15         driver.manage().window().maximize();
16         driver.manage().timeouts().implicitlyWait(Duration.ofSeconds(10));
17         //Casting with driver
18         JavascriptExecutor js=(JavascriptExecutor) driver;
19         //locate element for enter text
20         WebElement userName = driver.findElement(By.xpath("//input[@name='username']"));
21         WebElement password = driver.findElement(By.xpath("//input[@name='password']"));
22         //enter text using JavascriptExecutor
23         js.executeScript("arguments[0].value='Admin';",userName);
24         js.executeScript("arguments[0].value='admin123';", password);
25
26
27 }
```

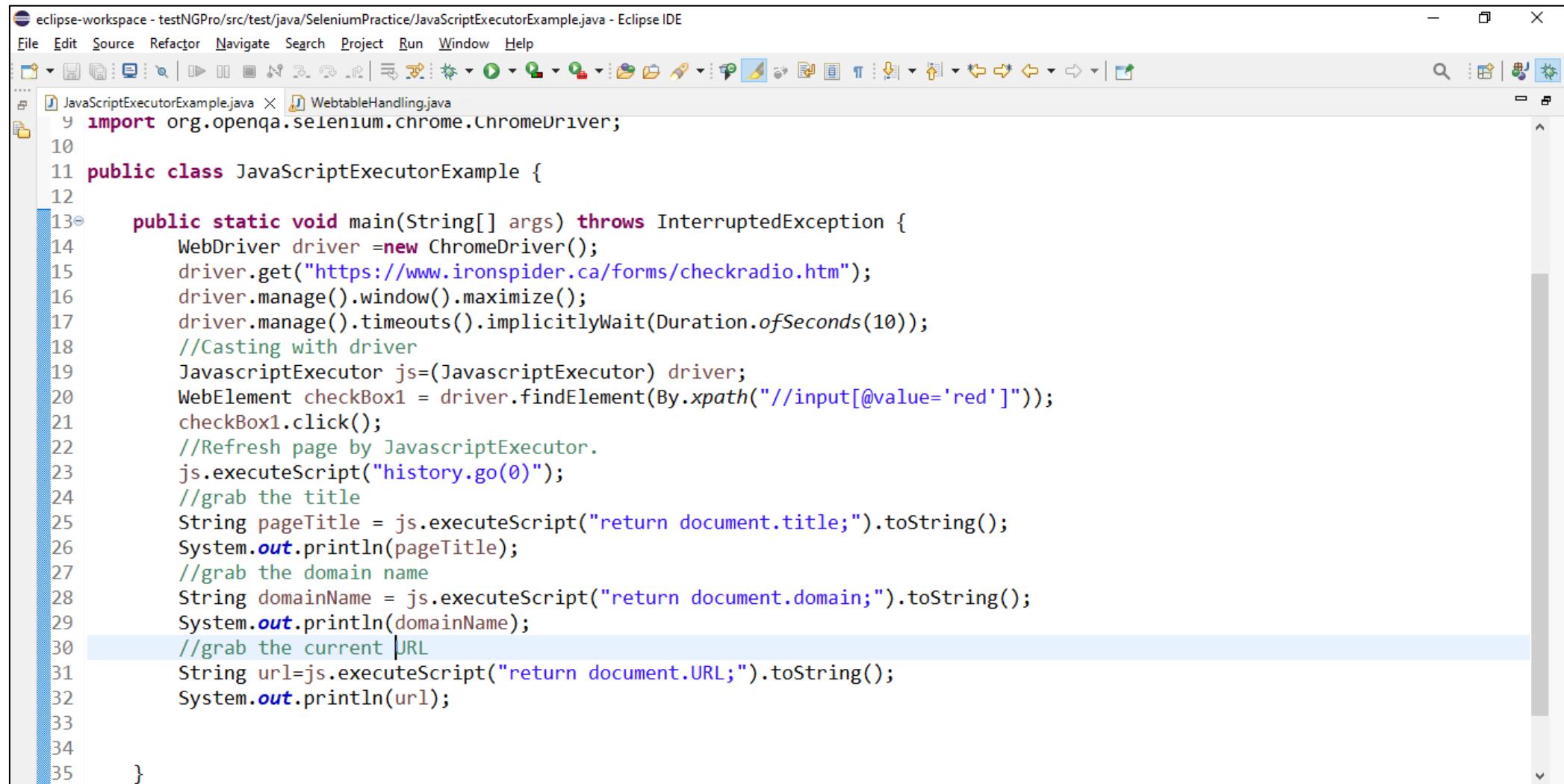
3. To generate Alert Pop window in Selenium Webdriver



The screenshot shows the Eclipse IDE interface with the title bar "eclipse-workspace - testNGPro/src/test/java/SeleniumPractice/JavaScriptExecutorExample.java - Eclipse IDE". The menu bar includes File, Edit, Source, Refactor, Navigate, Search, Project, Run, Window, and Help. The toolbar has various icons for file operations like Open, Save, Find, and Run. The package explorer shows two files: "JavaScriptExecutorExample.java" and "WebtableHandling.java". The Java code in "JavaScriptExecutorExample.java" is as follows:

```
1 package SeleniumPractice;
2
3 import java.time.Duration;
4
5 import org.openqa.selenium.JavascriptExecutor;
6 import org.openqa.selenium.WebDriver;
7 import org.openqa.selenium.chrome.ChromeDriver;
8
9 public class JavaScriptExecutorExample {
10
11     public static void main(String[] args) throws InterruptedException {
12         WebDriver driver =new ChromeDriver();
13         driver.get("https://opensource-demo.orangehrmlive.com/web/index.php/auth/login");
14         driver.manage().window().maximize();
15         driver.manage().timeouts().implicitlyWait(Duration.ofSeconds(10));
16         //Casting with driver
17         JavascriptExecutor js=(JavascriptExecutor) driver;
18         js.executeScript("alert('Welcome to JavaScriptExecutor !!')");
19
20     }
21
22 }
23
```

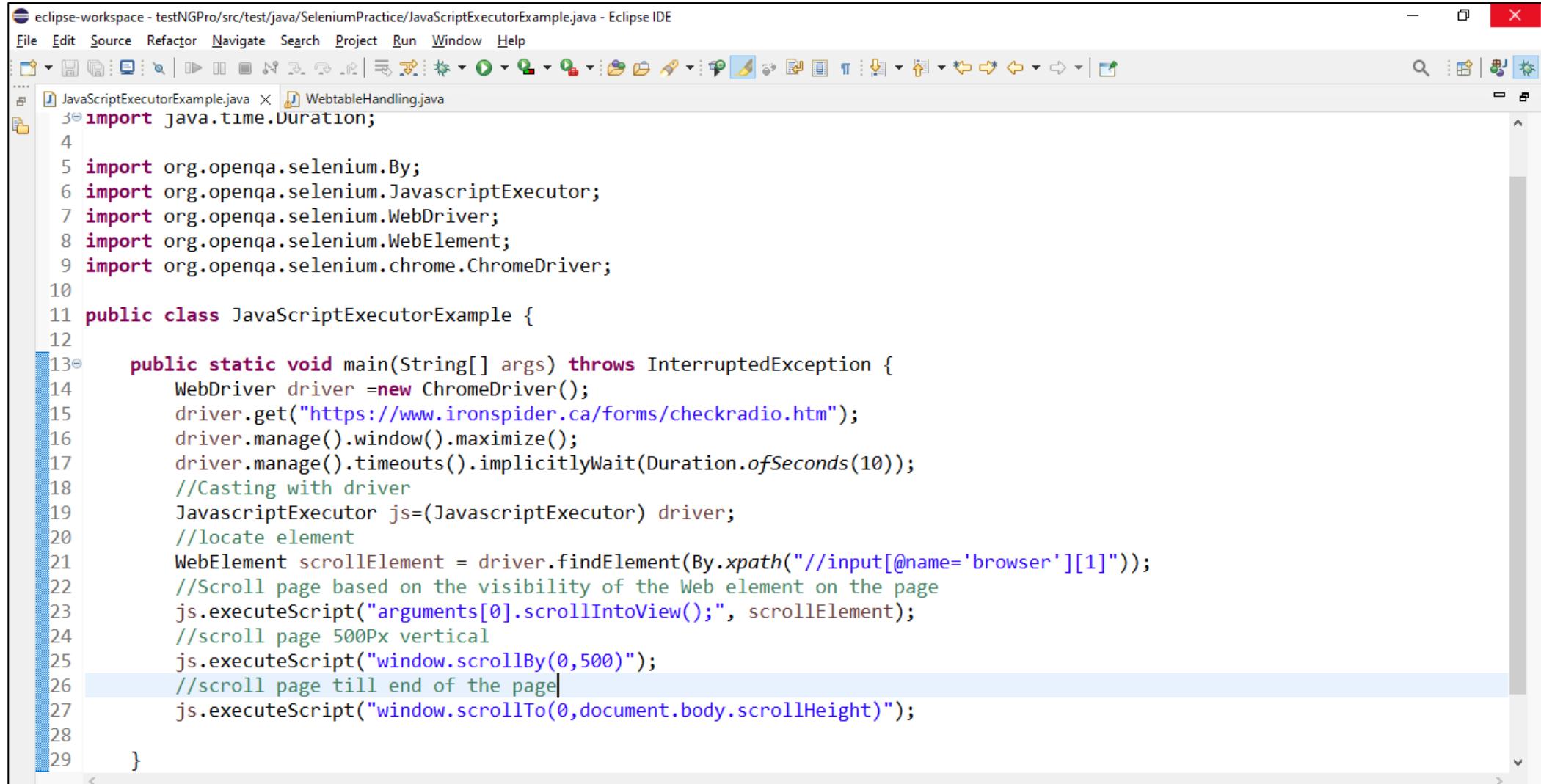
4. To get the Title, Domain name and URL of our webpage



The screenshot shows the Eclipse IDE interface with the title bar "eclipse-workspace - testNGPro/src/test/java/SeleniumPractice/JavaScriptExecutorExample.java - Eclipse IDE". The menu bar includes File, Edit, Source, Refactor, Navigate, Search, Project, Run, Window, and Help. The toolbar has various icons for file operations like Open, Save, Find, and Run. The left sidebar shows two open files: "JavaScriptExecutorExample.java" and "WebtableHandling.java". The main editor area contains the following Java code:

```
9 import org.openqa.selenium.chrome.ChromeDriver;
10
11 public class JavaScriptExecutorExample {
12
13     public static void main(String[] args) throws InterruptedException {
14         WebDriver driver =new ChromeDriver();
15         driver.get("https://www.ironspider.ca/forms/checkradio.htm");
16         driver.manage().window().maximize();
17         driver.manage().timeouts().implicitlyWait(Duration.ofSeconds(10));
18         //Casting with driver
19         JavascriptExecutor js=(JavascriptExecutor) driver;
20         WebElement checkBox1 = driver.findElement(By.xpath("//input[@value='red']"));
21         checkBox1.click();
22         //Refresh page by JavascriptExecutor.
23         js.executeScript("history.go(0)");
24         //grab the title
25         String pageTitle = js.executeScript("return document.title;").toString();
26         System.out.println(pageTitle);
27         //grab the domain name
28         String domainName = js.executeScript("return document.domain;").toString();
29         System.out.println(domainName);
30         //grab the current URL
31         String url=js.executeScript("return document.URL;").toString();
32         System.out.println(url);
33
34     }
35 }
```

5. To scroll the page vertically to the end and a specific web element.



The screenshot shows the Eclipse IDE interface with the title bar "eclipse-workspace - testNGPro/src/test/java/SeleniumPractice/JavaScriptExecutorExample.java - Eclipse IDE". The menu bar includes File, Edit, Source, Refactor, Navigate, Search, Project, Run, Window, and Help. The toolbar has various icons for file operations like Open, Save, Find, and Run. The left sidebar shows two open files: "JavaScriptExecutorExample.java" and "WebtableHandling.java". The main editor area contains Java code for a Selenium WebDriver example. The code imports necessary packages, initializes a ChromeDriver, navigates to a URL, and uses JavaScriptExecutor to scroll a specific element into view and then scroll to the bottom of the page.

```
1 import java.time.Duration;
2
3 import org.openqa.By;
4 import org.openqa.selenium.JavascriptExecutor;
5 import org.openqa.selenium.WebDriver;
6 import org.openqa.selenium.WebElement;
7 import org.openqa.selenium.chrome.ChromeDriver;
8
9 public class JavaScriptExecutorExample {
10
11     public static void main(String[] args) throws InterruptedException {
12         WebDriver driver =new ChromeDriver();
13         driver.get("https://www.ironspider.ca/forms/checkradio.htm");
14         driver.manage().window().maximize();
15         driver.manage().timeouts().implicitlyWait(Duration.ofSeconds(10));
16         //Casting with driver
17         JavascriptExecutor js=(JavascriptExecutor) driver;
18         //locate element
19         WebElement scrollElement = driver.findElement(By.xpath("//input[@name='browser'][1]"));
20         //Scroll page based on the visibility of the Web element on the page
21         js.executeScript("arguments[0].scrollIntoView();", scrollElement);
22         //scroll page 500Px vertical
23         js.executeScript("window.scrollBy(0,500)");
24         //scroll page till end of the page
25         js.executeScript("window.scrollTo(0,document.body.scrollHeight)");
26     }
27 }
```