Practical No. 2. FindElement() and FindElements() Methods in Selenium

Date: _____

Aim:

To find web elements using findElement() and findElements() methods.

Theory:

A web page is comprised of many different HTML elements, such as buttons, links, a body, labels, forms, and so on, that are named Web Elements in the context of WebDriver. Together, these elements on a web page will achieve the business functionality

There are different types of Web Elements such as <html>, <body>, <form>, <label>, <input>, and <a>, which together make a web page.

For example,

```
<label>Enter Username: </label>
```

Here, <label> is the start tag of the WebElement label. Enter Username: is the text present on the label element.

Similarly, take another WebElement:

```
<input type="text" name="Username"/>
```

Consider following HTML code

```
<html>
<body>
<form id="loginForm">
<label>Enter Username: </label>
<input type="text" name="Username"/>
<label>Enter Password: </label>
<input type="password" name="Password"/>
<input type="submit"/>
</form>
<a href="forgotPassword.html">Forgot Password ?</a>
</body>
</html>
```

In the preceding code, type and name are the attributes of the WebElement input with values text and Username, respectively.

Interaction with a web page requires a user to locate the web element.

There are multiple ways to uniquely identify a web element within the web page such as

- ID,
- Name,
- Class Name,
- Link Text,
- Partial Link Text,
- Tag Name
- XPATH.

findElement() and findElements() method

findElement() method is used to uniquely identify a (one) web element within the web page.

Whereas findElements() method is used to uniquely identify the list of web elements within the web page.

Syntax:

WebElement elementName = driver.findElement(By.LocatorStrategy("LocatorValue"));

The findElement() and By() methods instruct WebDriver to locate a WebElement on a web page, and once found, the findElement() method returns the WebElement instance of that element.

Actions such as click, type, and so on, are performed on a returned Web Element using the methods declared in the WebElement interface.

So, the input parameter for the findElement() method is the By instance.

The By instance is a WebElement-locating mechanism.

The return type of the findElement() method is the WebElement instance that represents the actual HTML element or component of the web page. The method returns the first WebElement that the driver comes across which satisfies the locating-mechanism condition. This WebElement instance will act as a handle to that component from then on. Appropriate actions can be taken on that component by the test script developer using this returned WebElement instance.

If WebDriver doesn't find the element, it throws a runtime exception named NoSuchElementException, which the invoking class or method should handle. The test script developer is advised to avoid using this method if he/she thinks the WebElement will not be present on the web page. For those purposes, we can use another method of WebDriver named findElements.

```
public class GoogleSearch {
public static void main(String[] args){
```

```
WebDriver driver = new FirefoxDriver();
driver.get("http://www.google.com");
WebElement searchBox = driver.findElement(By.name("q"));
searchBox.sendKeys("Finolex Academy");
searchBox.submit();
}
}
```

findElements() syntax:

FindElements in Selenium command takes in By object as the parameter and returns a list of web elements.

It returns an empty list if there are no elements found using the given locator strategy and locator value.

Below is the syntax of find elements command.

List<WebElement> elementName = driver.findElements(By.LocatorStrategy("LocatorValue"));

Example:

```
public class LocatingByTagname1 {
public static void main(String[] args) {
        System.setProperty("webdriver.gecko.driver","D:\\selenium_drivers\\geckodriver.exe");
        WebDriver driver = new FirefoxDriver();
        driver.get("http://demo.guru99.com/test/facebook.html");
        List <WebElement> list = driver.findElements(By.tagName("input"));
        for(int i = 0; i < list.size(); i++)
                {
                System.out.println(list.get(i).getAttribute("name"));
                }
        }
}
```

Implementation

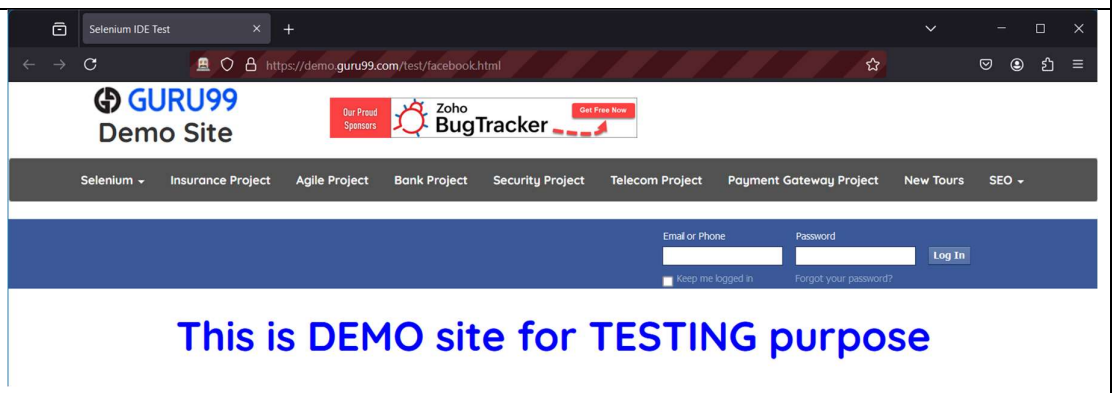
1	Locate all elements with tagname “input” on the given website http://demo.guru99.com/test/facebook.html
Code	**package pkg;** **import java.util.List;** **import org.openqa.selenium.By;** **import org.openqa.selenium.WebDriver;** **import org.openqa.selenium.WebElement;** **import org.openqa.selenium.firefox.FirefoxDriver;** **public class PracticalNo2A {** **public static void main(String[] args) {** **// TODO Auto-generated method stub** **// Setting Path to FireFox()** **System.setProperty("webdriver.gecko.driver","G:\\Selenium_setup\\geckodriver.exe");** **// Web driver for FireFox** **WebDriver driver = new FirefoxDriver();** **// Open Web Page Login.html** **driver.get("https://demo.guru99.com/test/facebook.html");** **// Locate elements** **List<WebElement> txtBoxes = driver.findElements(By.tagName("input"));** **// Display number of Web Elements having <input> tag and saved in list**

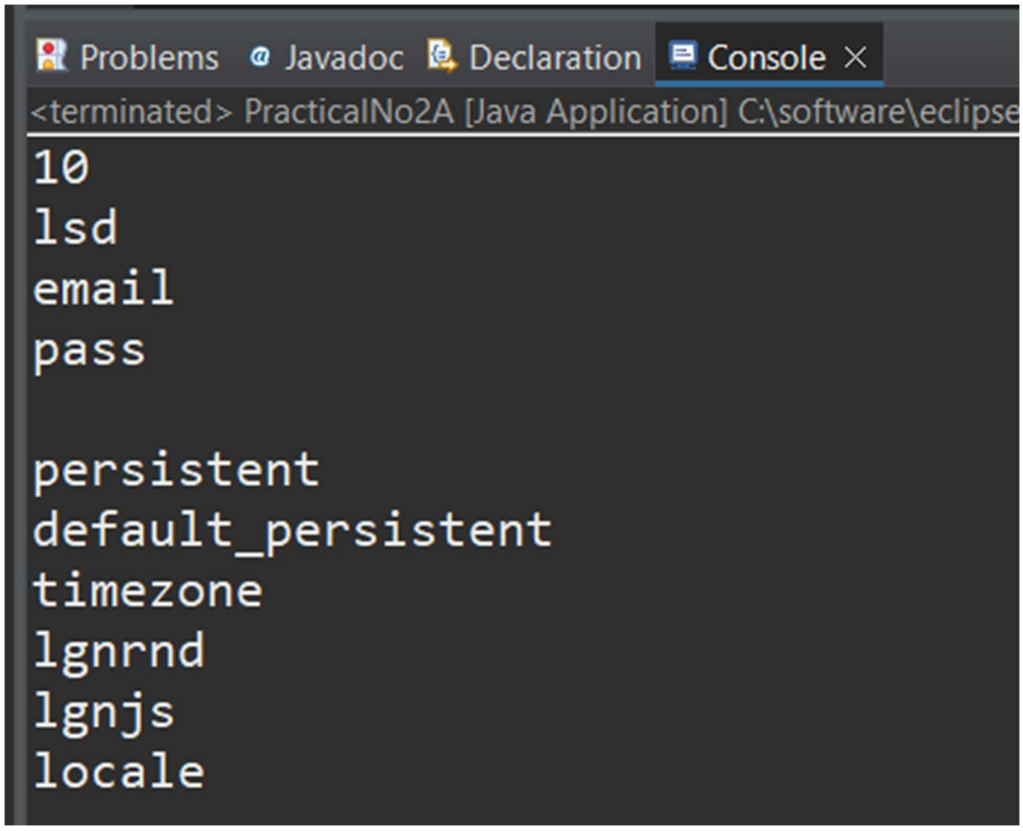
```
        System.out.println(txtBoxes.size());


        // Display value of name attribute in the List
        for(int i = 0; i < txtBoxes.size(); i++) {
                System.out.println(txtBoxes.get(i).getAttribute("name"));
        }


        }
}
```

Output



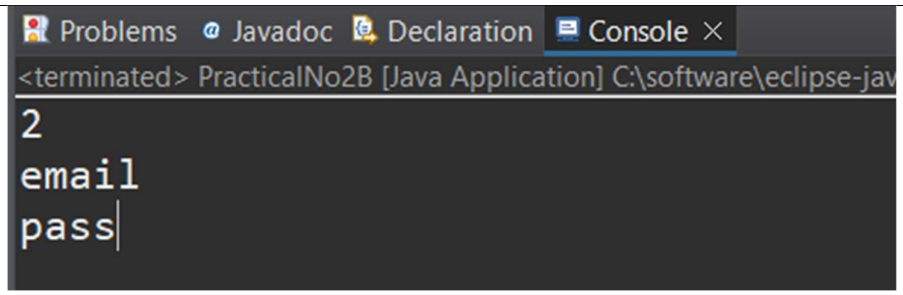


2	Locate all elements with classname “inputtext” on the given website http://demo.guru99.com/test/facebook.html
Code	**package pkg;** **import java.util.List;** **import org.openqa.selenium.By;** **import org.openqa.selenium.WebDriver;** **import org.openqa.selenium.WebElement;** **import org.openqa.selenium.firefox.FirefoxDriver;** **public class PracticalNo2B {** **public static void main(String[] args) {** **// TODO Auto-generated method stub** **// Setting Path to FireFox()** **System.*setProperty*("webdriver.gecko.driver","G:\\Selenium_setup\\geckodriver.exe");** **// Web driver for FireFox** **WebDriver driver = new FirefoxDriver();** **// Open Web Page Login.html** **driver.get("https://demo.guru99.com/test/facebook.html");** **// Locate elements** **List<WebElement> txtBoxes = driver.findElements(By.*className*("inputtext"));** **// Display number of Web Elements having <input> tag and saved in list** **System.*out*.println(txtBoxes.size());**

```
        // Display value of name attribute in the List
        for(int i = 0; i < txtBoxes.size(); i++) {
                System.out.println(txtBoxes.get(i).getAttribute("name"));
        }

        }
}
```

Output



Conclusion: Learnt to find single as well as multiple elements using findElement() and findElements().

After performing this Practical/lab, students are expected to answer following questions

Q.1. What is difference between findElement() and findElements()?