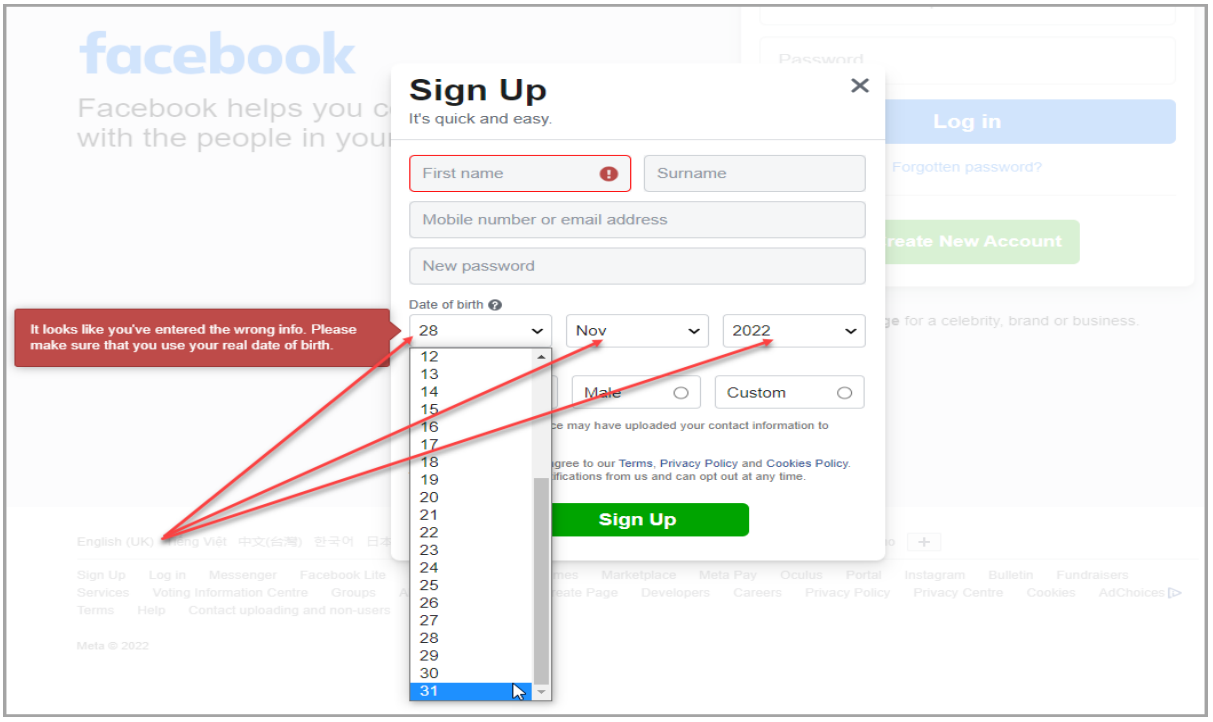
Trainer: LE DAC MINH
Phone: 0966152432
Email: ledacminh0305@gmail.com

Content

1. Default Dropdown list / Custom Dropdown list
2. User Interactions
3. Dropdown list/ User interactions/ Windows/ Tabs
4. Javascript Executor
5. Selenium WebDriver Wait

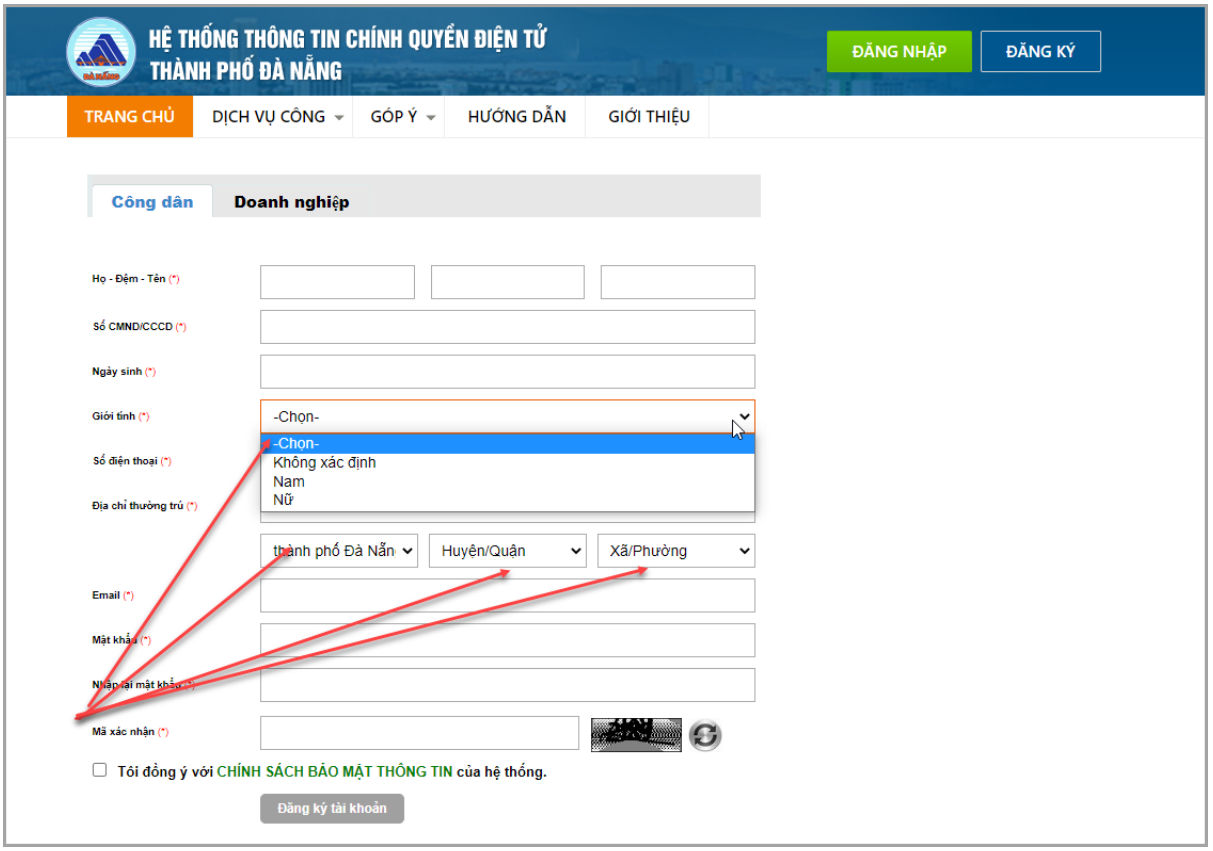
Trainer: LE DAC MINH
Phone: 0966152432
Email: ledacminh0305@gmail.com

Dropdown list



- https://www.facebook.com/

Trainer: LE DAC MINH
Phone: 0966152432
Email: ledacminh0305@gmail.com



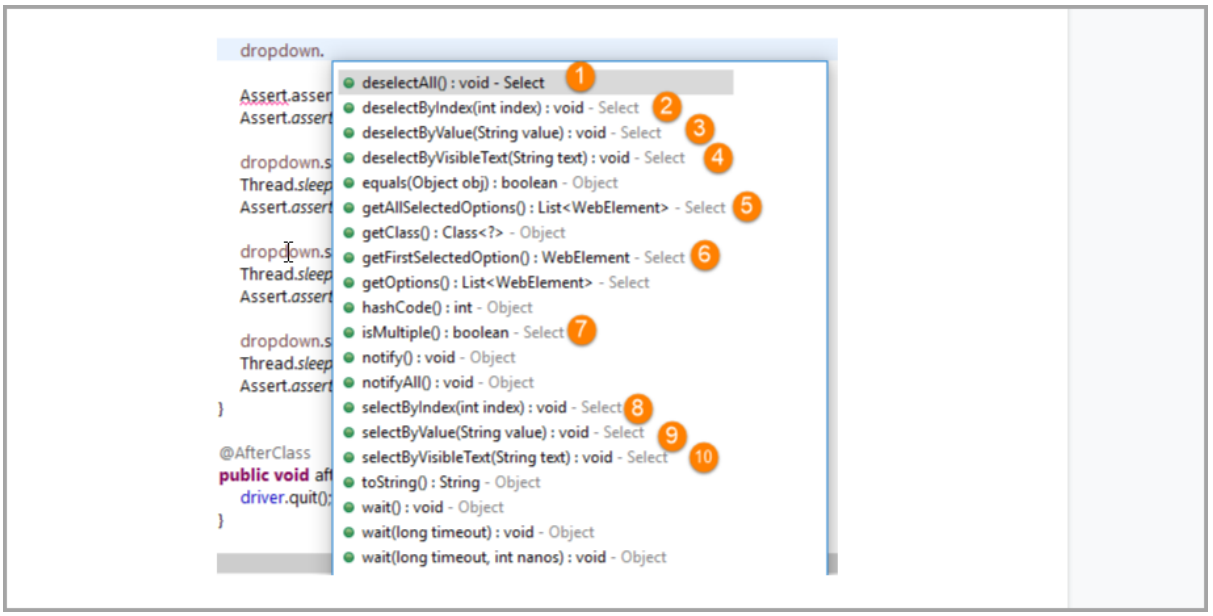
- https://egov.danang.gov.vn/reg

- How to check the one value in the dropdown to display correctly or not after selecting successfully.
- How to check the dropdown to be multi-select or single-select: multi-select is allowed to select many values in the dropdown.

Solution:

- Check all of the elements in the dropdown using Select Class instead of WeElement.
    - import org.openqa.selenium.support.ui.Select;
- Select Class provides methods and attributes to interact with the dropdown.

Trainer: LE DAC MINH
Phone: 0966152432
Email: ledacminh0305@gmail.com



- Select the value in the dropdown using **VisibleText**.
  - select.selectByVisiableText("Automation Tester");

- Select the value in dropdown using **value** .
  - select.selectByValue("manual");

- Select the value in the dropdown using **index.**
  - select.selectByIndex(3);

- How to check the dropdown does not support multi-select;
  - Assert.assertFalse(select.isMultiple());

- How to check the dropdown have 5 values.
  - Assert.assertEquals(5, select.getOptions().size());

- How to check the value is displayed correctly after selected successfully (the selected value is always displayed first).
  - Assert.assertEquals("Manual Tester", select.getFirstSelectedOption().getText());
- **Solution**
  - Click on the dropdown
  - Wait for all of the elements is displayed
  - Push all of the items in the list (List < WebElement>)
  - Using for loop pass each elements then getText()
  - If actualText = expectedText
    - scroll to that element
    - click on that element
    - exit the loop

Trainer: LE DAC MINH
Phone: 0966152432
Email: ledacminh0305@gmail.com

User Interactions

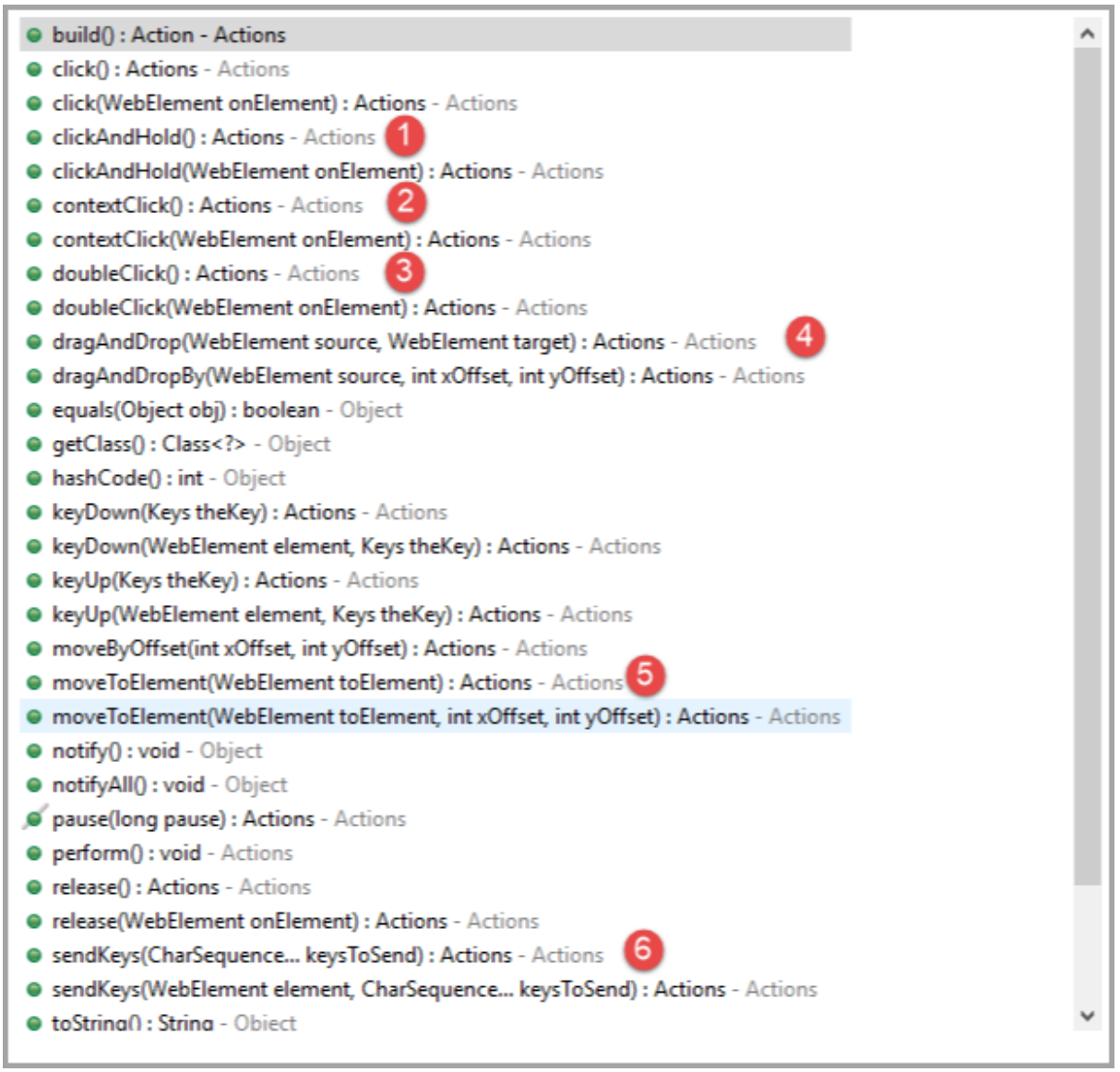
- **Mouse interactions**
  - click() - left click
  - doubleClick() - double click
  - clickAndHold() - press and hold
  - moveToElement() - hover
  - contextClick() - right click
  - dragAndDrop() - drag and drop

- **Keyboard interactions**
  - keyDown() / keyUp()

```
String osName = System.getProperty("os.name");
if(osName.toLowerCase().contains("mac os")) {
        action.keyDown(Keys.COMMAND).perform();
        } else {
        action.keyDown(Keys.CONTROL).perform();
}
```

- **SendKeys()**

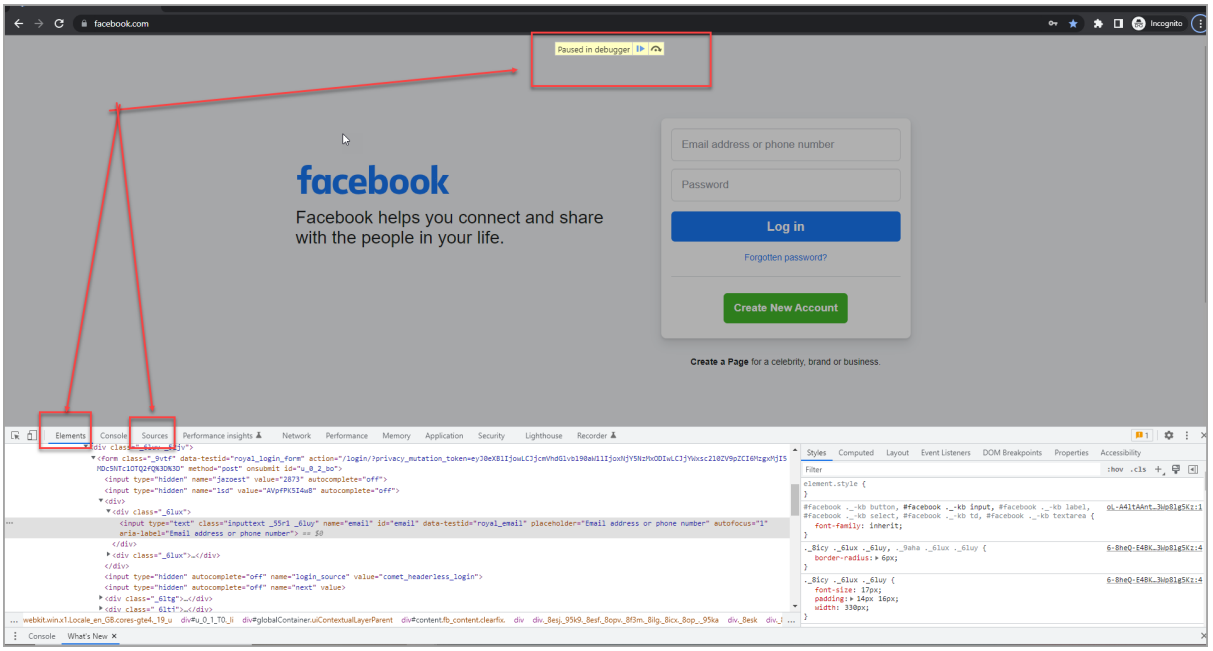
Trainer: LE DAC MINH
Phone: 0966152432
Email: ledacminh0305@gmail.com



- **import org.openqa.selenium.interactions.Actions();**
  - NOTE:
    - release: unhold left-click
    - perform: execute action - any methods need to perform()
    - moveToElement(...).perform()
    - contextClick(..).perform()
    - click(..).perform()

- **Tip to check element hover success**
  - Chrome (turn on debug)
    - Open Dev Tool - navigate to tab Sources
    - Hover on the element then press F8
    - Back to tab Elements and catch/ verify element

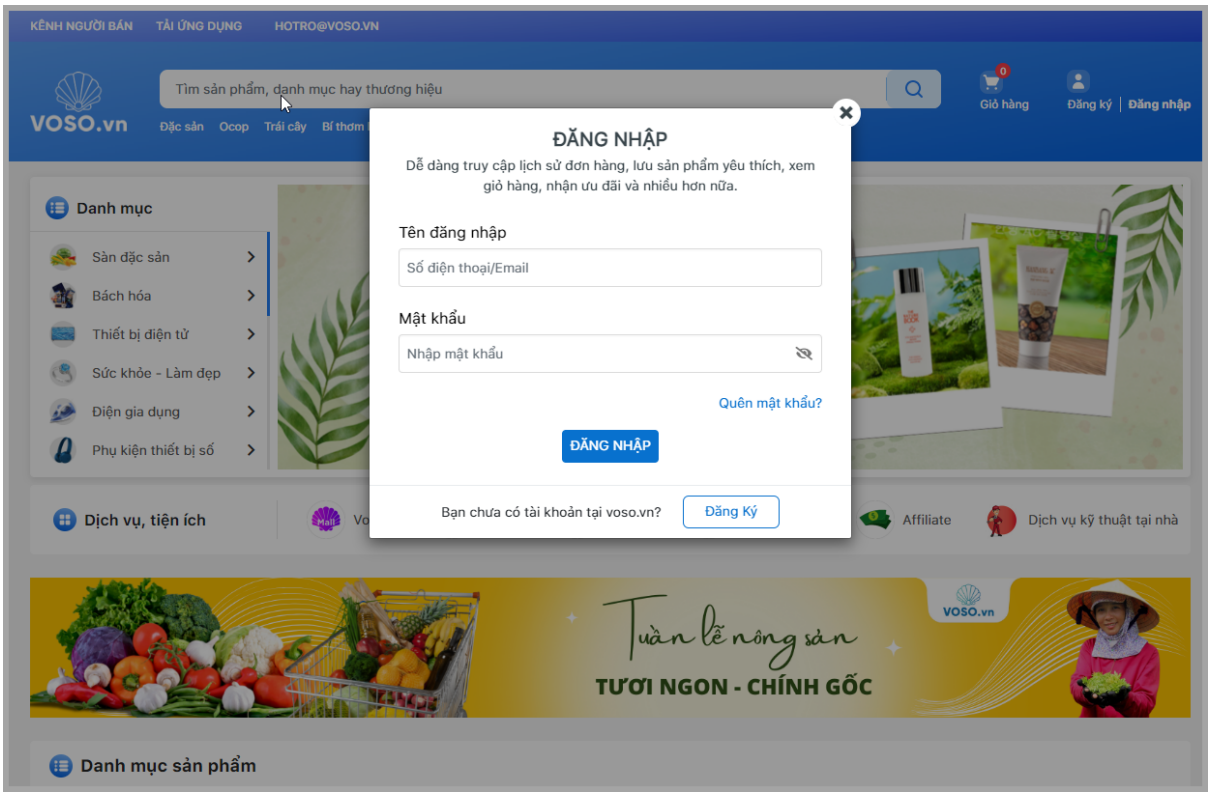
Trainer: LE DAC MINH
Phone: 0966152432
Email: ledacminh0305@gmail.com

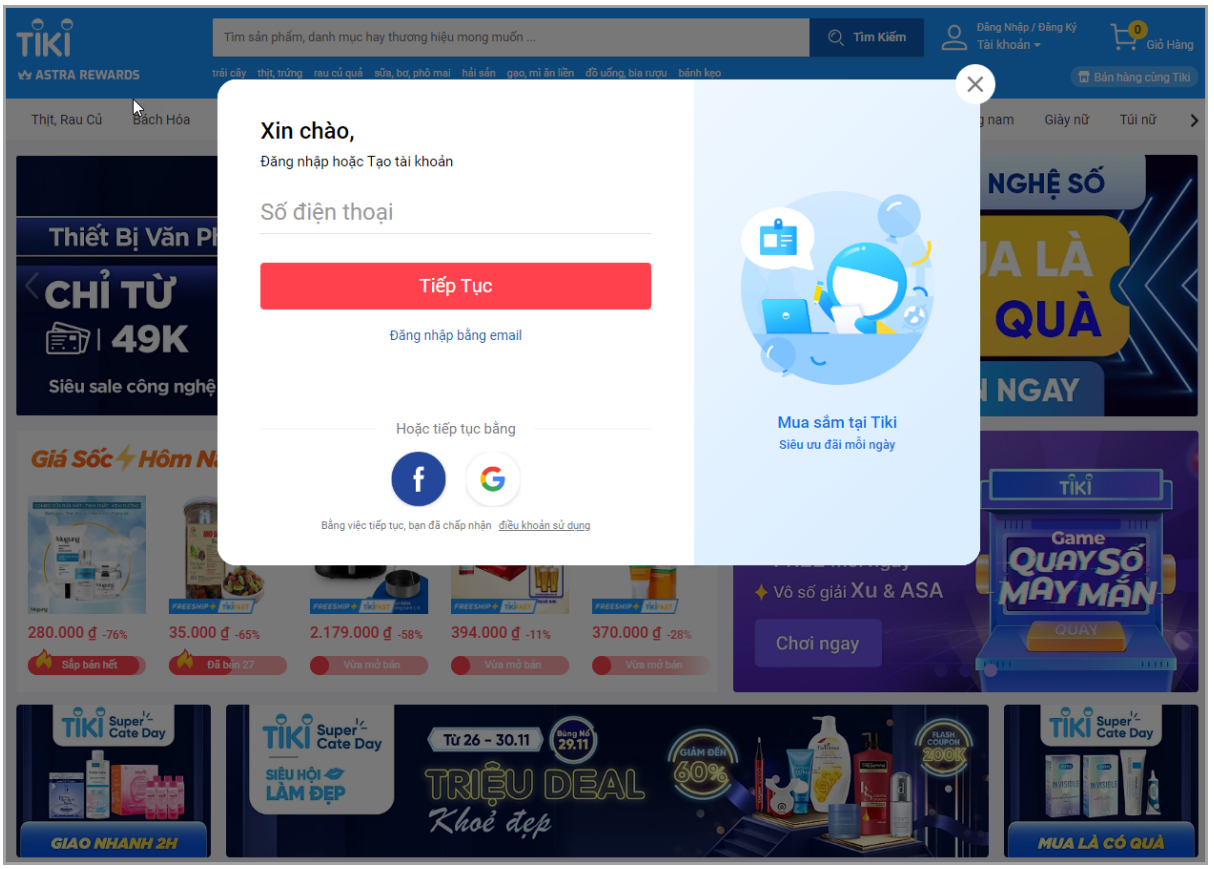


Popup/ Iframe/ Frame/ Windows/ Tab

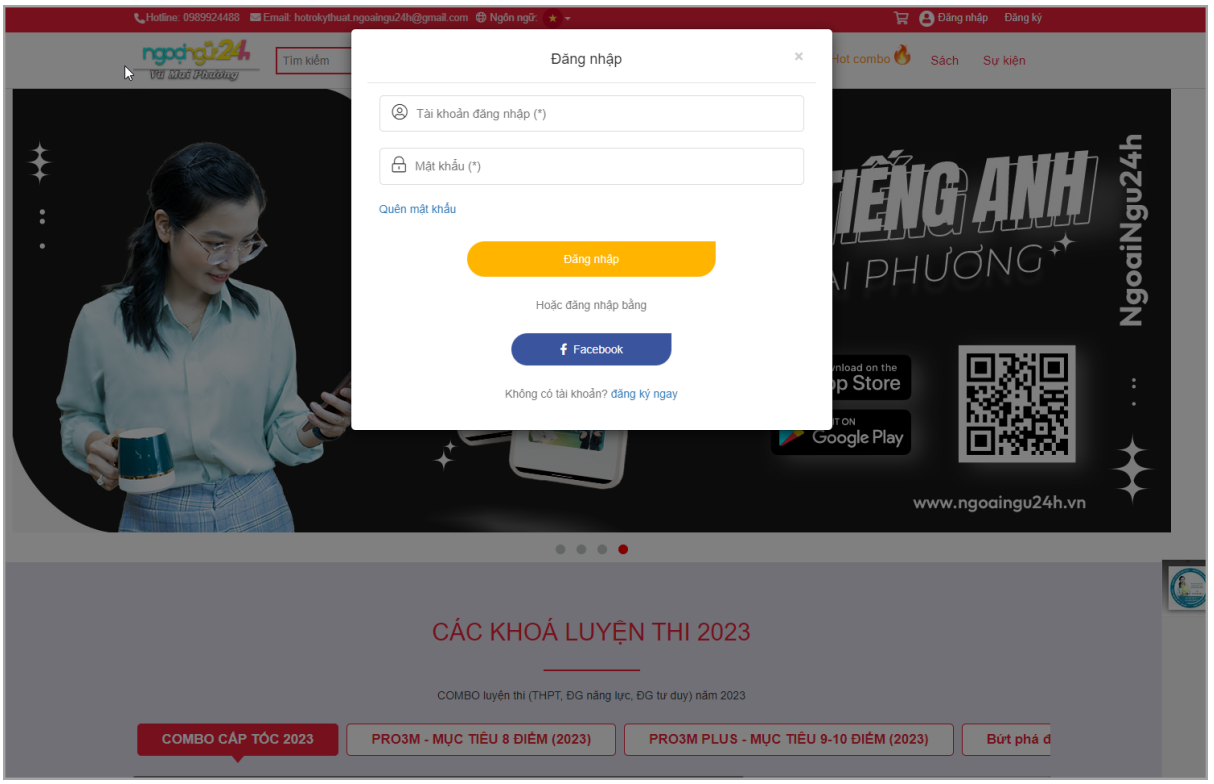
- **Popup**
  - https://voso.vn/#
    - Register/ Login
  - https://tiki.vn/
    - Register/ Login
  - https://ngoaingu24h.vn/
    - Register/ Login
  - https://skills.kynaenglish.vn/
    - Login

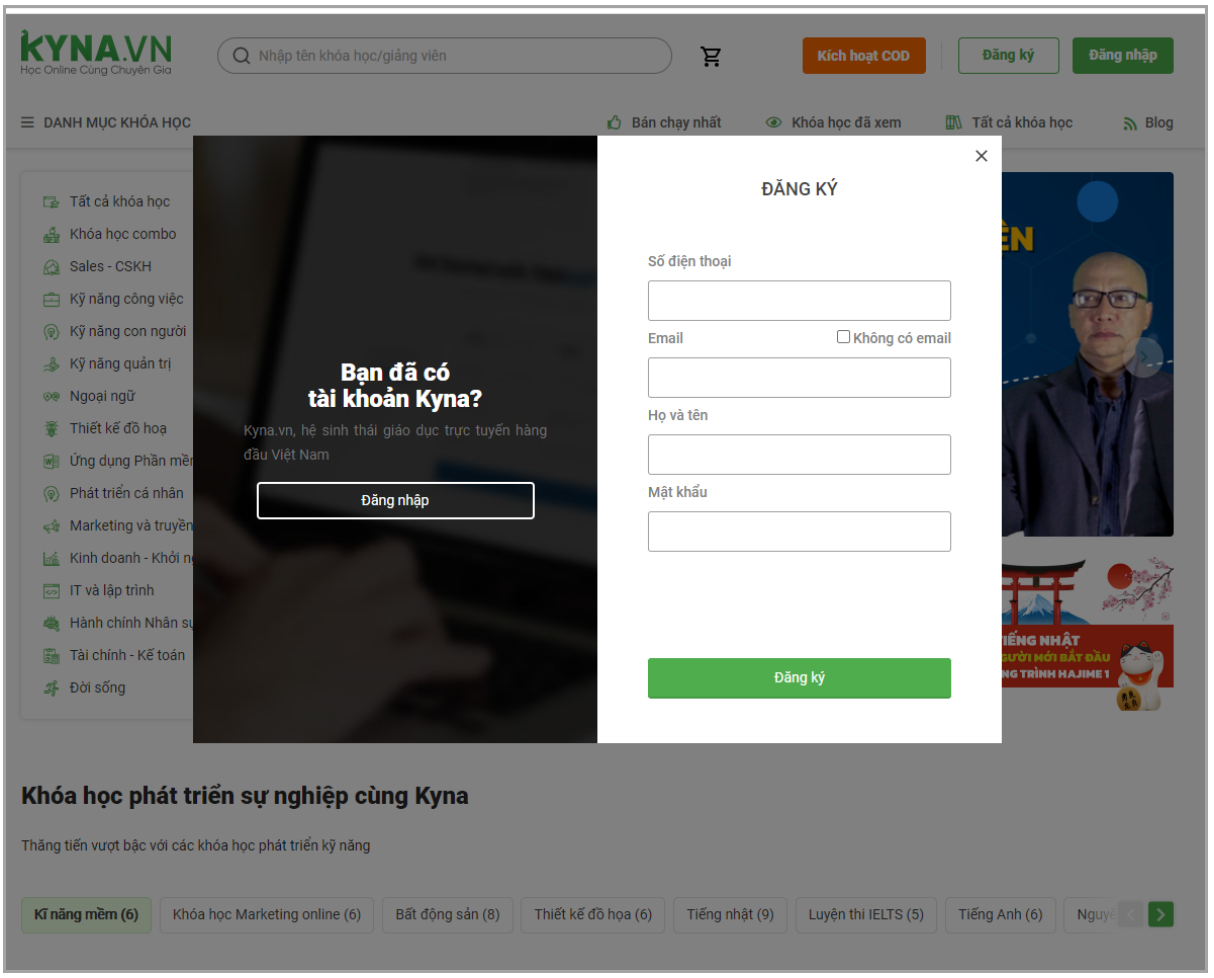
Trainer: LE DAC MINH
Phone: 0966152432
Email: ledacminh0305@gmail.com





Trainer: LE DAC MINH
Phone: 0966152432
Email: ledacminh0305@gmail.com





Trainer: LE DAC MINH
Phone: 0966152432
Email: ledacminh0305@gmail.com

- **Iframe/ Frame**
  - Cùng domain - > Frame
  - Khác domain - > Iframe
- **Switch to Frames by Index**
  - driver.switchTo().frame(0);
  - driver.quit();
- **Switch to Frames by WebElement**

```
WebElement iframe = driver.findElement(By.id("ifr_NewCereberusDetails_R1_Borrower"));
driver.switchTo().frame(iframe);
driver.quit();
```

- **Switching back to Main page from Frame**

```
WebElement iframe = driver.findElement(By.id("ifr_NewCereberusDetails_R1_Borrower"));
driver.switchTo().frame(iframe);
//Action
....
//Back to the main window
driver.switchTo().defaultContent();
driver.quit();
```

- **NoSuchElement Exeption in Frame**

```
//Step 01
driver.get("http://...");

//Step 02
//driver.switchTo().frame(iframe);
```

// Step 3: driver.findElement(By.xpath("//*[@id="firstname"]")).sendKeys("Le Dac Minh");

```
//Back to the main window
driver.switchTo().defaultContent();
driver.quit();
```

```
Error: org.openqa.selenium.NoSuchElementException: Unable to locate element:
{"method":"xpath","selector":"//*[@id='firstname']"}
```

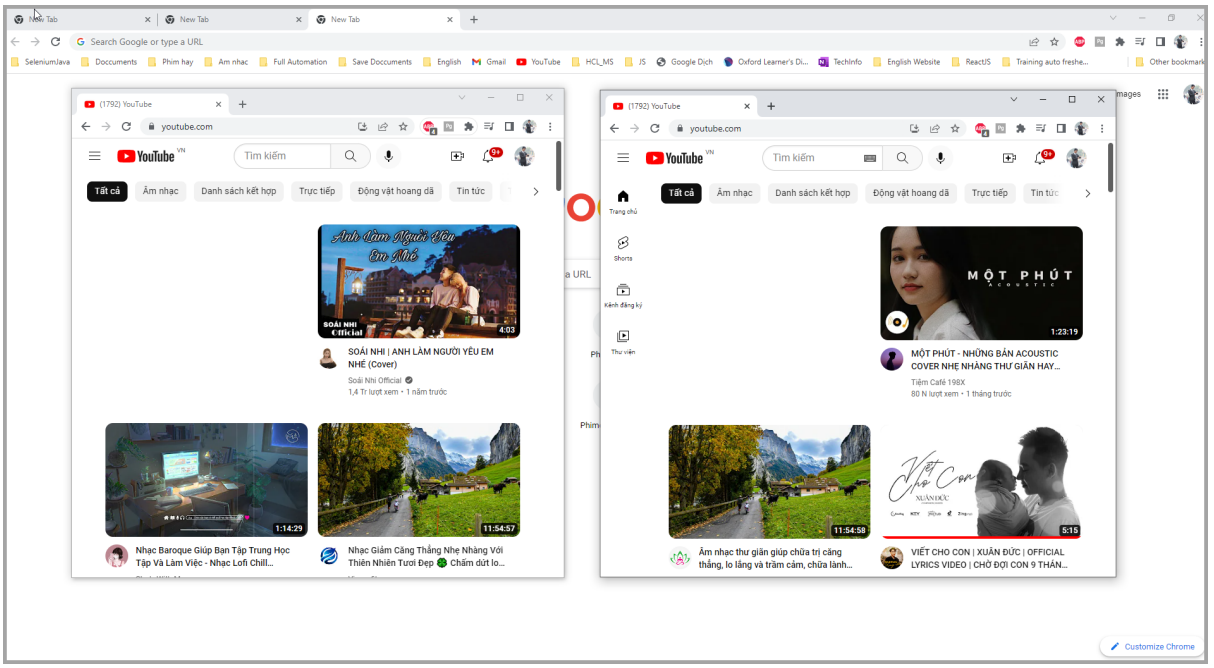
Trainer: LE DAC MINH
Phone: 0966152432
Email: ledacminh0305@gmail.com

- **NoSuchFrame Exception in Iframe**

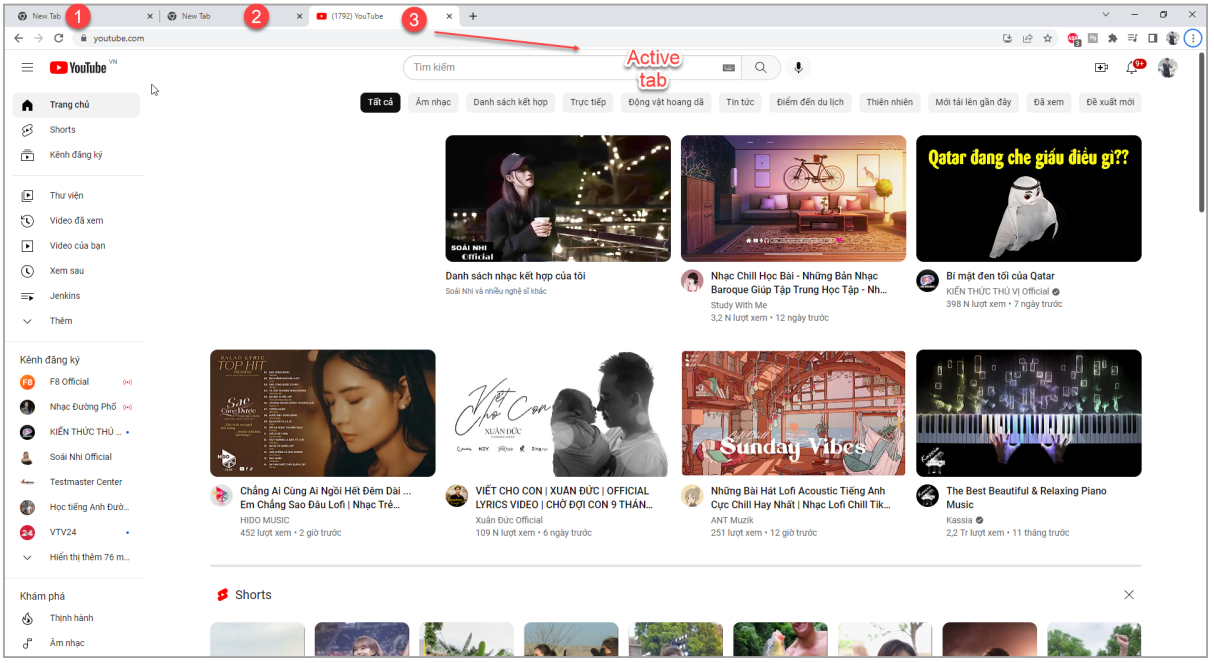
```
//Step 01
driver.get("http://...");

//Step 02
driver.switchTo().frame(0);

//Step 03
// driver.switchTo().defaultContent();

//Step 04
driver.switchTo().frame(1);
```

```
Error: org.openqa.selenium.NoSuchFrameException: No frame element found by name or id
```

- **Windows/ Tab**



Trainer: LE DAC MINH
Phone: 0966152432
Email: ledacminh0305@gmail.com



- **Case: Click on the link/ button … → Open a new tab/ windows**
- **Handle windows**

```
String parentWindow = driver.getWindowHandle();
=> returns the Window current focused window
3f94bb19-194f-41d8-b334-23130e31d275

Set <String> allWindows = driver.getWindowHandles();
=> returns set of all Window in browser session
3f94bb19-194f-41d8-b334-23130e31d275
aa93dc23-dfcc-4b1a-9183-a0de53b24e22
```

- **Switch to child windows (only 2 windows)**

```
public void switchToWindowByID(String parentID) {
        Set<String> allWindows = driver.getWindowHandles();
```

```
            for (String runWindow : allWindows) {
                if (!runWindow.equals(parentID)) {
                    driver.switchTo().window(runWindow);
                    break;
                }
            }
        }
```

- **Switch to child windows (greater than 2 windows and title of the pages are unique)**

```
public void switchToWindowByTitle(String title) {
            Set<String> allWindows = driver.getWindowHandles();
            for (String runWindows : allWindows) {
                driver.switchTo().window(runWindows);
                String currentWin = driver.getTitle();
                if (currentWin.equals(title)) {
                    break;
                }
            }
        }
```

- **Close all windows without parent window**

```
public boolean closeAllWindowsWithoutParent(String parentID) {
            Set<String> allWindows = driver.getWindowHandles();
            for (String runWindows : allWindows) {
                if (!runWindows.equals(parentID)) {
                    driver.switchTo().window(runWindows);
                    driver.close();
                }
            }
            driver.switchTo().window(parentID);
```

```
            if (driver.getWindowHandles().size() == 1)
                return true;
            else
                return false;
        }
```

Trainer: LE DAC MINH
Phone: 0966152432
Email: ledacminh0305@gmail.com

- **References:**
  - https://imalittletester.com/2020/05/21/working-with-windows-tabs-in-selenium/

Javascript Executor

- **Selenium Binding**
  - Selenium Java
  - Selenium C#
  - Selenium Python
  - Selenium Ruby
  - Selenium Javascript
- **Javascript Executor**
  - The library (Interface) of Selenium - lies in the package: org.openqa.selenium
  - Selenium is the library/tool to interact with browsers/elements. There are features that Selenium doesn't support while Javascript can do it by executing commands Javascript to the browser/ element.
  - Don't care about the status of the element displayed or not - only care whether the element is in DOM/ HTML or not.
- **Feature**
  - Search with big data
    - Click on the search button then load up to a thousand recordings.

- Click/ switch page
- Custom checkbox/ Radio button (Hidden Input)
- Scroll to the element
  - Custom dropdown list
  - Hidden element (to be covered by other elements)
  - One element is under one another element (not in the viewport) Firefox
- Remove an attribute in DOM
  - Date Picker (type = date) = > Textbox
  - Upload file on MAC OS
    - Remove attribute disabled (button upload)
- Demo for client/ team member - highlight element
- Get the value of hidden/ invisible element
  - Custom dropdown - Angular/ React dropdown
  - Custom dropdown - Editable dropdown
- Click to a hidden/ invisible element (ex: sub-menu/ link/ image…)

- **NOTE:**
  - We should limit using Javascript (trick) because we are able to handle by Selenium built-in. We should not abuse Javascript.
  - Selenium built-in = native user event (It is the common interaction of the End User)
  - We apply Javascript in Test Case. It can run fast. It will not wait for elements to be displayed before interacting.

- ## WebBrowser
- **Refresh the browser**

```
jsExecutor.executeScript("history.go(0)");
```

- **Get the inner text of the page**

```
String sText = jsExecutor.executeScript("return document.documentElement.innerText;").toString();
```

- **Get the title of the page**

```
String pageTitle = jsExecutor.executeScript("return document.title;").toString();
```

- **Get the domain of the page**

```
String domain = (String) jsExecutor.executeScript("return document.domain");
```

- **Scroll to 50 pixel**

```
jsExecutor.executeScript("window.scrollBy(0,50)");
```

Trainer: LE DAC MINH
Phone: 0966152432
Email: ledacminh0305@gmail.com

- **Scroll to the bottom of the page**

```
jsExecutor.executeScript("window.scrollBy(0,document.body.scrollHeight)");
```

- **Scroll down to pixel number**

```
jsExecutor.executeScript("window.scrollBy(0," + range + ")", "");
```

- **Navigate to the different page**

```
jsExecutor.executeScript("window.location = 'https://www.....'");
```

• WebElement

- **Click on an element**

```
WebElement element = driver.findElement(By.xpath("//abcdxyz"));
jsExecutor.executeScript("arguments[0].click();", element);
```

- **Scroll to an element**
  - true: above element
  - false: below element

```
WebElement element = driver.findElement(By.id("id_of_element"));
jsExecutor.executeScript("arguments[0].scrollIntoView(true);", element);
```

- **Hightlight Element**

```
jsExecutor.executeScript("arguments[0].style.border='2px groove green'", element);
```

- Remove an Attribute

```
jsExecutor.executeScript("arguments[0].removeAttribute('disabled');", element);
```

- **Get HTML5 validation message**

```
jsExecutor.executeScript("return arguments[0].validationMessage;", element);
```

- **Common functions**

Trainer: LE DAC MINH
Phone: 0966152432
Email: ledacminh0305@gmail.com

```java
public Object executeForBrowser(String javaScript) {
        return jsExecutor.executeScript(javaScript);
    }

    public String getInnerText() {
        return (String) jsExecutor.executeScript("return document.documentElement.innerText;");
    }

    public boolean areExpectedTextInInnerText(String textExpected) {
        String textActual = (String) jsExecutor.executeScript("return document.documentElement.innerText.match('" + textExpected + "')[0];");
        return textActual.equals(textExpected);
    }

    public void scrollToBottomPage() {
        jsExecutor.executeScript("window.scrollBy(0,document.body.scrollHeight)");
    }

    public void navigateToUrlByJS(String url) {
        jsExecutor.executeScript("window.location = '" + url + "'");
    }

    public void hightlightElement(String locator) {
        WebElement element = getElement(locator);
        String originalStyle = element.getAttribute("style");
        jsExecutor.executeScript("arguments[0].setAttribute('style', arguments[1])", element, "border: 2px solid red; border-style: dashed;");
        sleepInSecond(1);
        jsExecutor.executeScript("arguments[0].setAttribute('style', arguments[1])", element, originalStyle);
    }

    public void clickToElementByJS(String locator) {
        jsExecutor.executeScript("arguments[0].click();", getElement(locator));
    }

    public void scrollToElementOnTop(String locator) {
        jsExecutor.executeScript("arguments[0].scrollIntoView(true);", getElement(locator));
    }

    public void scrollToElementOnDown(String locator) {
        jsExecutor.executeScript("arguments[0].scrollIntoView(false);", getElement(locator));
    }

    public void sendkeyToElementByJS(String locator, String value) {
        jsExecutor.executeScript("arguments[0].setAttribute('value', '" + value + "')", getElement(locator));
    }

    public void removeAttributeInDOM(String locator, String attributeRemove) {
        jsExecutor.executeScript("arguments[0].removeAttribute('" + attributeRemove + "');", getElement(locator));
    }
```

Trainer: LE DAC MINH
Phone: 0966152432
Email: ledacminh0305@gmail.com

```java
public void hightlightElement(String locator) {
    WebElement element = getElement(locator);
    String originalStyle = element.getAttribute("style");
    jsExecutor.executeScript("arguments[0].setAttribute('style', arguments[1])", element, "border: 2px solid red; border-style: dashed;");
    sleepInSecond(1);
    jsExecutor.executeScript("arguments[0].setAttribute('style', arguments[1])", element, originalStyle);
}

public void clickToElementByJS(String locator) {
    jsExecutor.executeScript("arguments[0].click();", getElement(locator));
}

public void scrollToElementOnTop(String locator) {
    jsExecutor.executeScript("arguments[0].scrollIntoView(true);", getElement(locator));
}

public void scrollToElementOnDown(String locator) {
    jsExecutor.executeScript("arguments[0].scrollIntoView(false);", getElement(locator));
}

public void sendkeyToElementByJS(String locator, String value) {
    jsExecutor.executeScript("arguments[0].setAttribute('value', '" + value + "')", getElement(locator));
}

public void removeAttributeInDOM(String locator, String attributeRemove) {
    jsExecutor.executeScript("arguments[0].removeAttribute('" + attributeRemove + "');", getElement(locator));
}

public String getElementValidationMessage(String locator) {
    return (String) jsExecutor.executeScript("return arguments[0].validationMessage;", getElement(locator));
}

public boolean isImageLoaded(String locator) {
    boolean status = (boolean) jsExecutor.executeScript(
            "return arguments[0].complete && typeof arguments[0].naturalWidth != 'undefined' && arguments[0].naturalWidth > 0",
            getElement(locator));
    return status;
}

public WebElement getElement(String locator) {
    return driver.findElement(By.xpath(locator));
}
```

Selenium WebDriver Wait

- **Element status**
  - Visible
  - Invisible
  - Presence
  - Staleness
- **Wait**
  - Implicit Wait
  - Explicit Wait
  - Fluent Wait
- **findElement vs findElements**
  - don’t find the element
  - has one element

- more than one element