**CÁC HÀM CƠ BẢN CỦA WEBDRIVER**

**I. Các hàm định vị phần tử .**

Để định vị các phần tử trong Selenium Webdriver, chúng ta có thể sử dụng các phương thức dưới đây:

* **findElement() hoặc findElementBy...()** : Trả về một đối tượng WebElement với điều kiện đầu vào
* **FindElements() hoặc findElementsBy...()**: Tìm kiếm tất cả các phần tử đáp ứng điều kiện và sau đó trả về một danh sách các đối tượng thuộc WebElement. ( java.util.List<WebElement>)

|  |  |  |
| --- | --- | --- |
| **Loại** | **Phương thức** | **Mô tả** |
| ID | driver.findElement(By.id(<element ID>)) | Định vị một phần tử bằng thuộc tính ID |
| driver.findElementById(<element ID>) |
| Name | driver.findElement(By.name(<element name>)) | Định vị một phần tử bằng thuộc tính Name |
| driver.findElementByName(<element name>) |
| Class  Name | driver.findElement(By.className(<element class>) | Định vị một phần tử bằng thuộc tính Class |
| driver.findElementByClassName(<element class>) |
| Tag Name | driver.findElement(By.tagName (<htmltagname>)) | Định vị một phần tử bằng thẻ Tag HTM |
| driver.findElementByTagName(<htmltagname>) |
| CSS | driver.findElement(By.cssSelector (<css selector>) | Định vị một phần tử bằng CSS |
| driver.findElementByCssSelector(<css selector>) |
| XPath | driver.findElement(By.xpath (<xpath>)) | Định vị một phần tử bằng truy vấn XPath |
| driver.findElementByXPath(<xpath>) |
| Link | driver.findElement(By.linkText (<linktext>) | Định vị phần tử bằng link văn bản liên kết |
| driver.findElementByLinkText(<linktext>) |
| Một phần Link | driver.findElement(By.partialLinkText (<linktext>) | Định vị phần tử bằng một phần link văn bản liên kết |
| driver.findElementByPartialLinkText(<linktext>) |

Một số ví dụ:

|  |
| --- |
| WebElement element1 = **driver**.findElementById(**"i\_agree"**);  WebElement element2 = **driver**.findElementByXPath(**"//a[text()='Sign In']"**);  List<WebElement> elements = **driver**.findElementsByClassName(**"txtinput"**); |

**II. Cách khởi tạo driver.**

|  |  |
| --- | --- |
| **Trình duyệt** | **Cú pháp** |
| Chrome | **import** org.openqa.selenium.chrome.ChromeDriver;  ….. WebDriver driver = **new** ChromeDriver(); |
| FireFox | **import** org.openqa.selenium.firefox.FirefoxDriver; …... WebDriver driver = **new** FirefoxDriver(); |
| InternetExplorer (IE) | **import** org.openqa.selenium.ie.InternetExplorerDriver; …... WebDriver driver = **new** InternetExplorerDriver(); |
| Safari | **import** org.openqa.selenium.safari.SafariDriver; …... WebDriver driver = **new** SafariDriver() |

**III. Các thao tác cơ bản với driver.**

|  |  |
| --- | --- |
| **Cú pháp** | **Mô tả** |
| **driver**.get(**"https://www.google.com/"**); | Tải một trang web |
| **driver**.navigate().to(**"https://www.google.com/"**); | Tải hoặc điều hướng đến trang web |
| **driver**.navigate().back(); | Quay về trang trước đó |
| **driver**.navigate().refresh(); | Làm mới trang hiện tại |
| **driver**.navigate().forward(); | Điều hướng đến trang tiếp theo sau |
| String title = **driver**.getTitle(); | Lấy ra title của trang |
| String url = **driver**.getCurrentUrl(); | Lấy ra đường dẫn trang hiện tại |
| String html = **driver**.getPageSource(); | Lấy về mã HTML |

**# Xử lý với của sổ và tab trình duyệt.**

Trong nhiều trường hợp, khi chúng ta click chuột vào hình ảnh, một đường dẫn hay một thành phần trên trang web hiện tại thì nó sẽ mở ra một tab mới hoặc một cửa sổ mới.  
Vậy làm thế nào để có thể làm việc với các Tab và các cửa sổ này?

Mỗi một cửa sổ, tab sẽ được đinh danh bởi một ID : String

|  |  |
| --- | --- |
| **Cú pháp** | **Mô tả** |
| String handle = **driver**.getWindowHandle(); | Lấy ra ID của cửa sổ hiện tại |
| Set<String> handle = **driver**.getWindowHandles();  -**Set** : à một danh sách tương tự như ArrayList, tuy nhiên khác với ArrayList, các phần tử trong Set là duy nhất (không có 2 phần tử trùng nhau)  <java.util.Set> | lấy ra danh sách các cửa sổ đang chạy từ driver (chú ý với cửa sổ mà người dụng tự mở thì selenium không thể lấy được) |

Vd: chuyển sang tab thứ 2 , lấy ra button Submit

|  |
| --- |
| Set<String> windowHandles = **driver**.getWindowHandles(); String[] tagIds = (String[]) windowHandles.toArray(); **if** (tagIds.**length** > 2) {  String firstTab = tagIds[0]; *//Tab đầu* String secondTab = tagIds[1]; *//Tab thứ hai* **driver**.switchTo().window(secondTab);  WebElement element = **driver**.findElementById(**"submit-btn"**);  element.click(); } |

**# Xử lý thời gian chờ tải trang.**

**Vd:** Đợi tối đa 30s để chờ trang được tải xong, sau đó mới thực hiện các thao tác tiếp theo.

|  |
| --- |
| **driver**.manage().timeouts().pageLoadTimeout(30, TimeUnit.***SECONDS***); |

**IV. Các thao tác cơ bản với WebElement.**

* **clear() :** Dùng để xóa nội dung của một ô text.

|  |
| --- |
| WebElement element = **driver**.findElementById(**"email"**); element.clear(); |

* **click() :** Nhấp chuột vào phần tử

|  |
| --- |
| WebElement element = **driver**.findElementById(**"ok"**); element.click(); |

* **sendKeys() :** Nhập text/chuỗi

|  |
| --- |
| WebElement element = **driver**.findElementById(**"email"**); element.sendKeys(**"example@gmail.com"**); |

* **isEnable() :** Kiểm tra phần tử có được enable hay không

|  |
| --- |
| WebElement element = **driver**.findElementById(**"ok"**); **if**(element.isEnabled()) {  element.click(); } |

* **isDisplayed() :** Kiểm tra phần tử có được hiển thị trên trang web hay không.  
   Trên thực tế, do phân quyền khác nhau nên với mỗi tài khoản thì sẽ tương ứng hiển thị các trường khác nhau trên website.

|  |
| --- |
| WebElement element = **driver**.findElementById(**"upload"**); **if**(element.isDisplayed()) {  *//the action should only show to admin, guest not allow* } |

* **isSelected() :** Kiểm tra phần tử có được chọn hay không (checkbox, radio button)

|  |  |
| --- | --- |
|  | WebElement element = **driver**.findElementById(**"agree"**); **if**(element.isSelected()) {   } |

* **submit() :** gửi biểu mẫu , hoặt động tốt hơn click() nếu đây là một biểu mẫu (Form).

Nếu việc gửi biểu mẫu là trang hiện tại thay đổi thì phương thức này sẽ đợt cho đến khi trang web mới được tải.

|  |
| --- |
| WebElement element = **driver**.findElementById(**"submit-btn"**); element.submit(); |

* **getText() :** Lấy ra chuỗi text đang được phần tử hiển thị

|  |
| --- |
| WebElement element = **driver**.findElementByName(**"description"**); String content = element.getText(); |

* **getTagName() :**  Lấy ra tên thẻ Tag HTML của phần tử
* **getCssValue(<thuộc tính css>) :**  Lấy ra giá trị thuộc tính CSS của phần tử
* **getAttribute(<Thuộc tính attribute>) :** Lấy ra giá trị thuộc tính attribute của phần tử
* **getSize() :** Lấy ra chiều rộng, chiều cao của phần thử

|  |
| --- |
| WebElement element = **driver**.findElementById(**"submit-btn"**); Dimension dimensions = element.getSize(); System.***out***.println(**"Height: "** + dimensions.Height +  **" Width: "** + dimensions.Width); |

* **getLoaction() :** Lấy về vị trí tọa độ x,y của phần tử trên trang.

|  |
| --- |
| WebElement element = **driver**.findElementById(**"submit-btn"**); Point point = element.getLocation(); System.***out***.println(**"X : "** + point.X +  **"Y : "** + point.Y); |