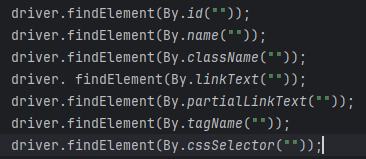
# **Selenium Web Driver Locators**

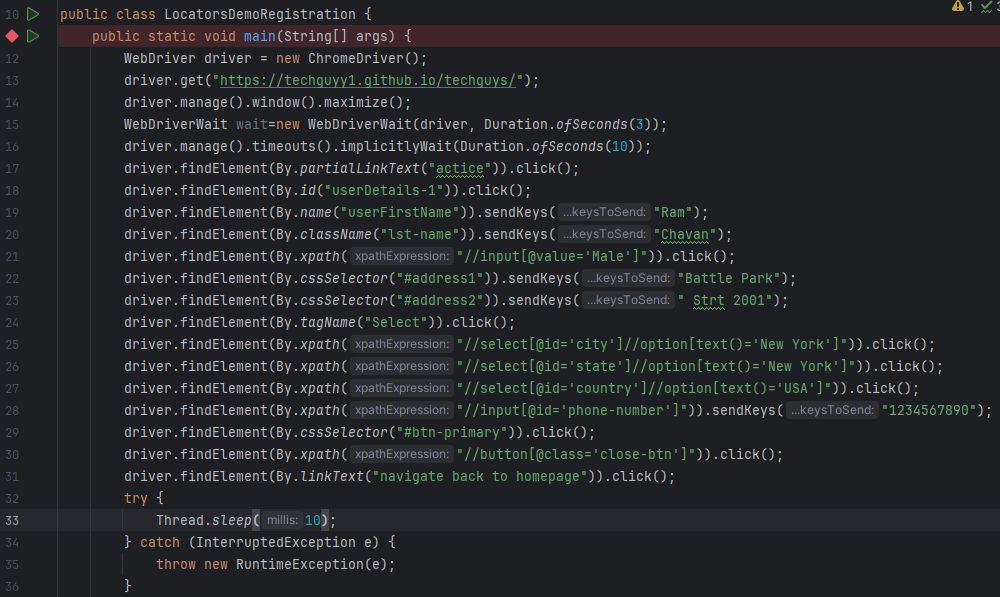
* As a part of automation, selenium performs actions such as click, enter input on the web page. Hence to perform that action we need to locate the element on webpage.
* By using **locators**, we can identify the web element on the HTML webpage. Selenium Web Driver uses any of the following locator to identify the web element on the webpage.

Selenium provides support for below 8 traditional location strategies in WebDriver:

**Types of locators**

|  |  |  |
| --- | --- | --- |
| Locator | Description | Example |
| id | Locates elements whose ID attribute matches the search value |  |
| class name | Locates elements whose class name contains the search value |  |
| name | Locates elements whose NAME attribute matches the search value |  |
| tag name | Locates elements whose tag name matches the search value |  |
| css selector | Locates elements matching a CSS selector |  |
| link text | Locates anchor elements whose visible text matches the search value |  |
| Partial link text | Locates anchor elements whose visible text contains the search value. If multiple elements are matching, only the first one will be selected. |  |
| xpath | Locates elements matching an XPath expression |  |





* **Key points:**

**driver->** It is an object or instance of ChromeDriver() class created using the Webdriver interface.

**findElement ->** is a WebDriver method

**By ->** Predefined class in selenium

**Types of Xpath**

1. **Absolute Xpath**

It’s a direct way of finding an element on webpage, it’s a full path from the root of the HTML document to the input field. It relies on the exact structure of the webpage, making it longer and more prone to breaking if the structure changes.

**Example**

//\*[@id="navbar"]/ul/li[3]/a

1. **Relative Xpath**

It’s a technique to find the element on webpage by giving precise path from the HTML document root. With the help of tag and associated attribute we can create absolute xpath in selenium. It will directly locate/search element on the web page by its tag name and associated attributes.

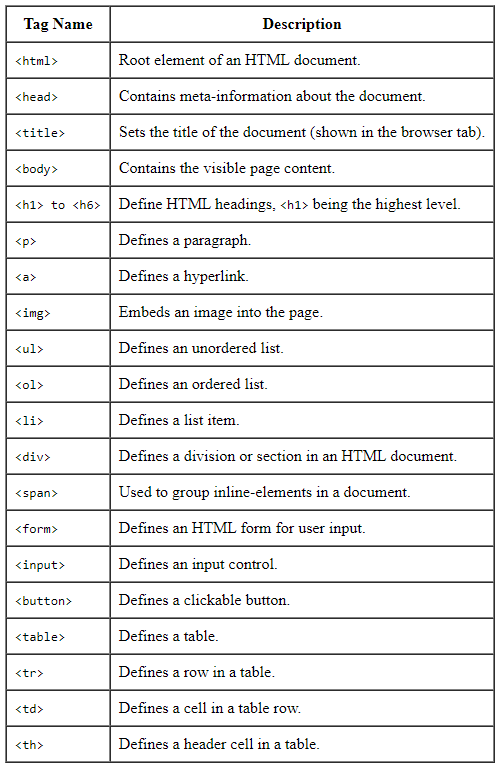
**Syntax:**



**Example:**



\*\* HTML Tag list:

****