

# Selenium with Java

## Lecture 5

### CSS Selector in Selenium:

CSS (Cascading Style Sheets) Selectors in Selenium are used to identify and locate web elements based on their id, class, name, attributes and other attributes. CSS is a preferred locator strategy as it is simpler to write and faster as compared to XPath.

`By.cssSelector(String cssSelector)` method is used to locate the elements in Selenium WebDriver. This method accepts a CSS Selector String as an argument which defines the selection method for the web elements.

The CSS Selector combines an element selector and a selector value that can identify particular elements on a web page. CSS selectors can locate web elements without ID, class, or Name.

### Types of CSS Selectors in Selenium

1. ID
2. Class
3. Attribute
4. Sub-String
5. Inner String

**1 ID:** In CSS, we can use “#” notation to select the “id” attribute of an element.

StackDemo

Offers Orders Favourites

a#offers.Navbar\_link\_3Blki 79.13 x 21

Vendors: 25 Product(s) found.

Order by Select

Apple Samsung  
Google OnePlus

iPhone 12 iPhone 12 Mini iPhone 12 Pro Max iPhone 12 Pro

Elements Console Sources Network Performance Memory Application Security Lighthouse Recorder Performance insights

Styles Computed Layout Event Listener

```
<div class="Navbar_root__2kbI9">
  <div class="mx-auto max-w-8xl px-6">
    <div class="flex justify-between align-center flex-row py-4 md:py-6 :py-6 relative"> <flex>
      <div class="flex flex-1 items-center"> <flex>
        <a class="Navbar_logo_2655Y" href="/"></a>
        <nav class="space-x-4 ml-6" style="display: flex;"> <flex>
          ... <a id="offers" class="Navbar_link_3Blki" href="/offers"><flex>
            <strong>Offers</strong>
          </a>
        </flex>
      </div>
    </div>
  </div>
</div>
```

a#offers

```
driver.findElement(By.cssSelector("a#offers"));
```

**2. Class :** In CSS, we can use “.” notation to select the “class” attribute of an element.

StackDemo

Offers Orders Favourites Sign

a.Navbar\_logo\_2655Y 257 x 45

iPhone 11 iPhone 11 Pro iPhone XS iPhone XR

\$ 599.00 \$ 699.00 \$ 549.00 \$ 499.00

or 8 x \$ 74.88 or 3 x \$ 233.00 or 4 x \$ 137.25 or 5 x \$ 99.80

Elements Console Sources Network Performance Memory Application Security Lighthouse Recorder Performance insights

Styles Computed Layout Event Listener

```
<div class="mx-auto max-w-8xl px-6">
  <div class="flex justify-between align-center flex-row py-4 md:py-6 :py-6 relative"> <flex>
    <div class="flex flex-1 items-center"> <flex>
      <a class="Navbar_logo_2655Y" href="/"></a>
      <div width="257" height="45" viewBox="0 0 257 45" fill="none" value="https://www.w3.org/2000/svg">
        <g filter="url(#filter0_d)"></g> Use $0 in the console to refer to this element.
        <path d="M190.008 29.3796H195.362C196.544 29.3796 197.543 29.1282 198.358 28.6254C199.18 28.1575 199.822 27.5639 200.283
          26.8446C200.456 26.5722 200.604 26.3103 200.727 26.0589C200.842 25.8005 200.929 25.5072 200.987 25.179C201.109 24.5295 201.17 23.3842 201.17 21.743C201.17 20.1996 201.131 19.1137 201.051 18.4852C200.95 17.8566 200.709 17.256 200.327 16.6834
        </path>
      </div>
    </div>
  </div>
</div>
```

a.Navbar\_logo\_2655Y

```
driver.findElement(By.cssSelector(".Navbar_root__2kbI9"));
```

3. Attribute: Apart from “id” and “class”, other attributes can also be used to locate web elements using CSS selector.

The screenshot shows a browser window for 'StackDemo' at 'bstackdemo.com'. The navigation bar includes 'Offers', 'Orders', and 'Favourites'. The 'Favourites' tab is highlighted with a blue background. A tooltip over the 'Favourites' tab displays the CSS selector 'a#favourites.Navbar\_link\_3Bk1'. The DOM tree in the developer tools shows the 'Favourites' tab's element structure, and the bottom status bar shows the selector 'a[href="/favourites"]'. A red arrow points from the explanatory text below to the 'Favourites' tab in the DOM tree.

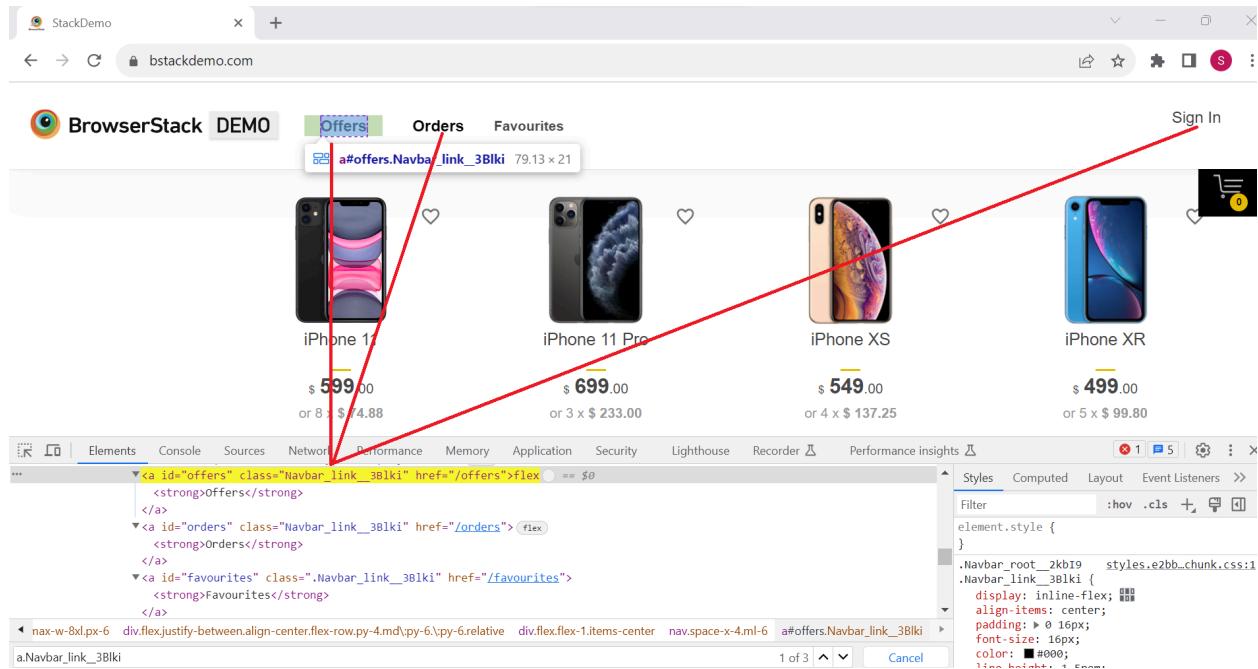
For the WebElement “Favourites” tab, tag name is “a” and href value is “/favourites”.

```
driver.findElement(By.cssSelector("a[href='/favourites']"));
```

4. Combining Attributes: From above examples we understood how we can uniquely identify elements using CSS selectors, however sometimes, using only class/ id/ single attribute does not yield a unique locator for a given web element. In such cases, we can combine multiple attributes to fetch a unique locator.

- Class and attribute example:

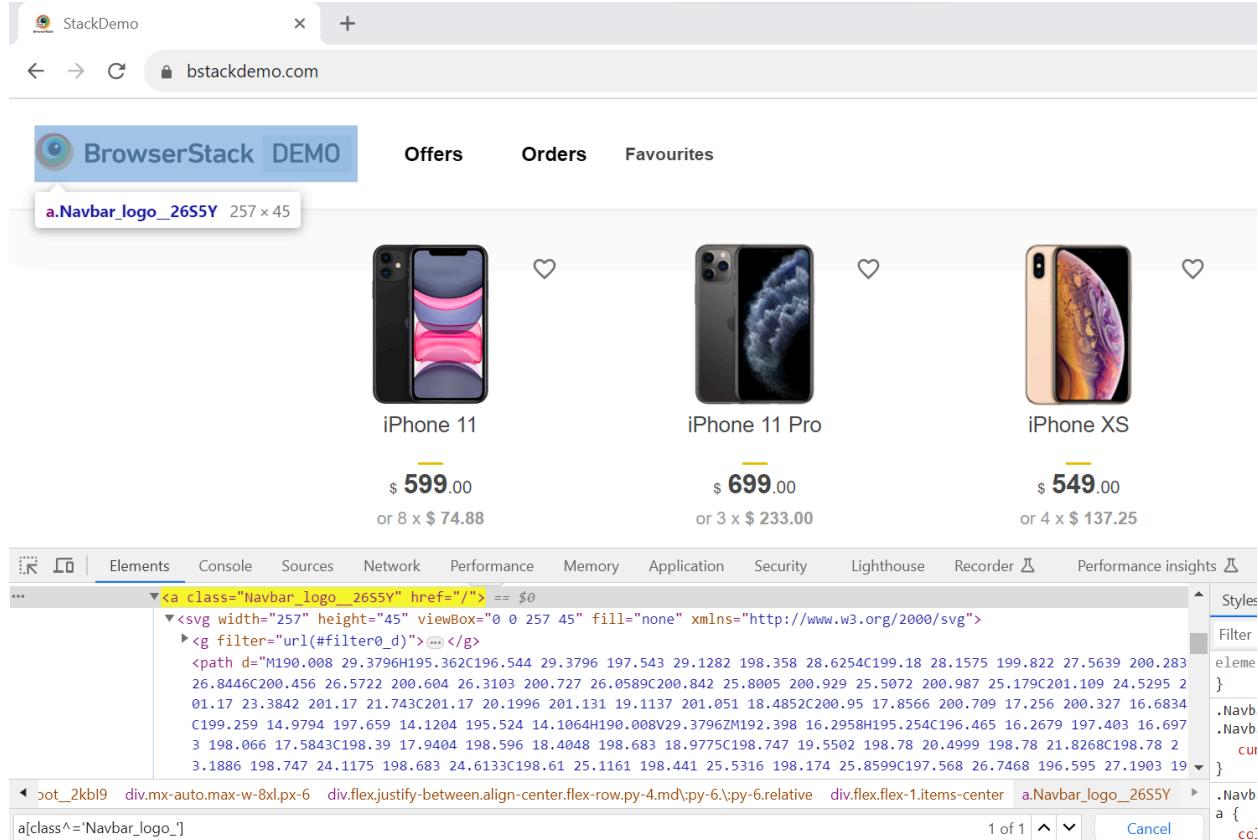
If we want to locate the WebElement “Offers” tab by just class value, it gives 3 results, which means it is not unique and pointing to 3 web elements on DOM. To make it unique we should also use the “href” attribute.



```
driver.findElement(By.cssSelector("a.Navbar_link_3Blki[href='/orders']"));
```

5. **SubString:** CSS Selectors in Selenium allows to match a partial String with the help of various symbols to represent the start, end and contents inside a text.

- **Matching a prefix (Starts with: ^):** Locate the web element using the substring that starts with a certain value.



`driver.findElement(By.cssSelector("a[class^='Navbar_logo_']"));`

- **Matching a suffix (Ends with: \$):** Locate the web element using the substring that ends with a certain value.

Screenshot of a web browser showing a product listing for three iPhone models. The page title is "StackDemo" and the URL is "bstackdemo.com". The navigation bar includes "Offers", "Orders", and "Favourites". The products listed are:

- iPhone 11**: \$599.00 (or 8 x \$74.88)
- iPhone 11 Pro**: \$699.00 (or 3 x \$233.00)
- iPhone XS**: \$549.00 (or 4 x \$137.25)

The "Elements" tab of the developer tools is selected, showing the HTML structure for the logo element. The class "a.Navbar\_logo\_26S5Y" is highlighted.

```

<a class="Navbar_logo_26S5Y" href="/" == $0
  <svg width="257" height="45" viewBox="0 0 257 45" fill="none" xmlns="http://www.w3.org/2000/svg">
    <g filter="url(#filter0_d)"></g>
      <path d="M190.008 29.3796H195.362C196.544 29.3796 197.543 29.1282 198.358 28.6254C199.18 28.1575 199.822 27.5639 200.283 26.8446C200.456 26.5722 200.604 26.3103 200.727 26.0589C200.842 25.8005 200.929 25.5072 200.987 25.179C201.109 24.5295 201.17 23.3842 201.17 21.743C201.17 20.1996 201.131 19.1137 201.051 18.4852C200.95 17.8566 200.709 17.256 200.327 16.6834 C199.259 14.9794 197.659 14.1204 195.524 14.1064H190.008V29.3796ZM192.398 16.2958H195.254C196.465 16.2679 197.403 16.697 3 198.066 17.5843C198.39 17.9404 198.596 18.4048 198.683 18.9775C198.747 19.5502 198.78 20.4999 198.78 21.8268C198.78 2 3.1886 198.747 24.1175 198.683 24.6133C198.61 25.1161 198.441 25.5316 198.174 25.8599C197.568 26.7468 196.595 27.1903 19
  </g>
</svg>

```

```
driver.findElement(By.cssSelector("a[class$='26S5Y']"));
```

- **Matching a substring (contains: \*):** Locate the web element by matching the substring.

Screenshot of a web browser showing a demo page for "BrowserStack DEMO". The page displays three iPhone models: iPhone 11, iPhone 11 Pro, and iPhone XS, each with a price of \$599.00 and payment options. The "iPhone 11" section is highlighted with a yellow box around its logo element.

The browser's developer tools are open, specifically the Elements tab. The selected element is `a class="Navbar_logo_26S5Y" href="/" == $0`. The element is an anchor tag with a class of "Navbar\_logo\_26S5Y" and a href attribute of "/". It contains an SVG icon with a width of 257 and a height of 45. The SVG includes a path definition with various coordinates.

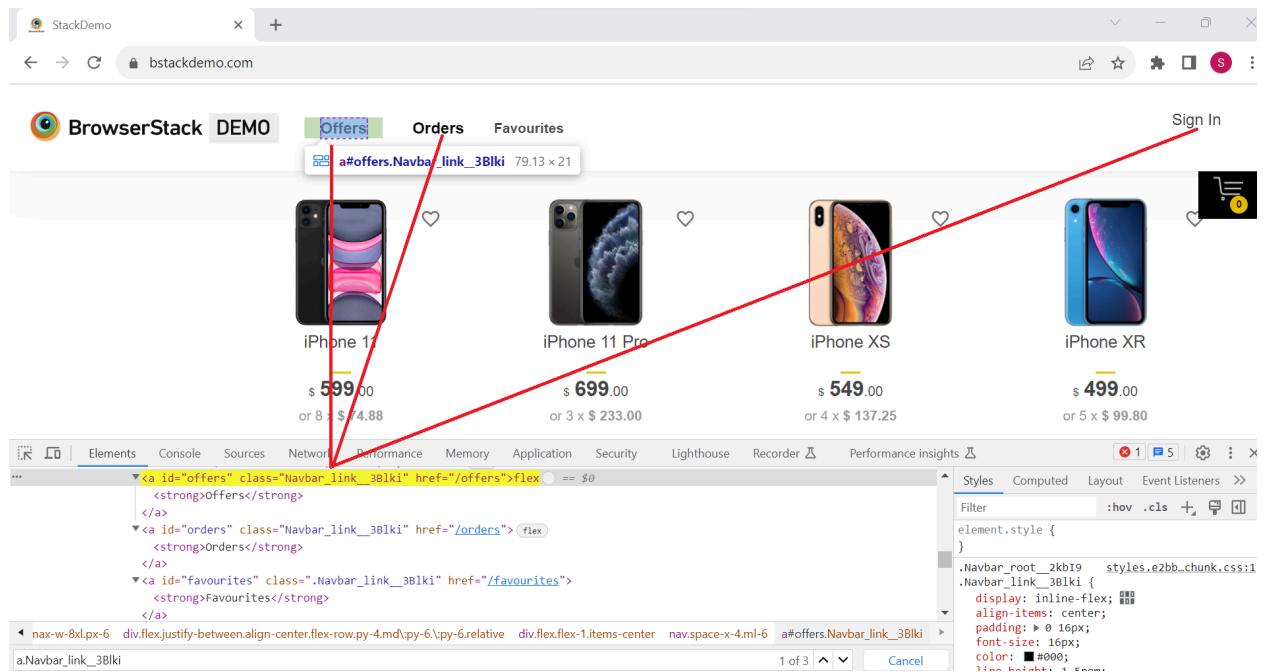
```

<a class="Navbar_logo_26S5Y" href="/" == $0
  <svg width="257" height="45" viewBox="0 0 257 45" fill="none" xmlns="http://www.w3.org/2000/svg">
    <g filter="url(#filter0_d)"></g>
      <path d="M190.008 29.3796H195.362C196.544 29.3796 197.543 29.1282 198.358 28.6254C199.18 28.1575 199.822 27.5639 200.283 26.8446C200.456 26.5722 200.604 26.3103 200.727 26.0589C200.842 25.8005 200.929 25.5072 200.987 25.179C201.109 24.5295 201.17 23.3842 201.17 21.743C201.17 20.1996 201.131 19.1137 201.051 18.4852C200.95 17.8566 200.709 17.256 200.327 16.6834 C199.259 14.9794 197.659 14.1204 195.524 14.1064H190.008V29.3796ZM192.398 16.2958H195.254C196.465 16.2679 197.403 16.697 3 198.066 17.5843C198.39 17.9404 198.596 18.4048 198.683 18.9775C198.747 19.5502 198.78 20.4999 198.78 21.8268C198.78 23.1886 198.747 24.1175 198.683 24.6133C198.61 25.1161 198.441 25.5316 198.174 25.8599C197.568 26.7468 196.595 27.1903 19
  </svg>

```

The bottom status bar of the developer tools shows the current file is "a[class\*='logo\_']".

```
driver.findElement(By.cssSelector("a[class*='logo']"));
```



## Finding child or Subchild Elements:

**Direct Child:** child combinator (>) is used to select direct child

Syntax : tagName[AttributeName='AttributeValue']>tagName[AttributeName='AttributeValue']

Example: *select#CompanyCountry-Kmm7>option*

Job title

Work email

Company

Country/Region

India

United States

Afghanistan

Albania

Algeria

American Samoa

Andorra

Anguilla

Antarctica

Antigua And Barbuda

Argentina

Armenia

I agree to the Main Services Agreement – Developer S acknowledge, as described in the Developer Document autonomous and other generative AI features; and (2) org, and may terminate any org that has been inactive

We value your privacy. To learn more, visit our Privacy Statement

I'm not a robot

reCAPTCHA

Privacy - Terms

32°C Sunny

Google Chrome isn't your default browser Set as default

Elements Console Sources Network

Select an element in the page to inspect it - Ctrl+Shift+C. ainer section" >

<div class="fields-container section"> ... </div>

<div class="fields-wrapper clearfix"> ... </div>

<div class="field-container-50" ... </div>

<div class="field-container-50" ... </div>

<div class="country-state-group section"> ... </div>

<div class="entry-wrap section"> ... </div>

<div class="country\_field selectFieldInput" ... </div>

<div class="field valid" ... </div>

<select id="CompanyCountry-Kmm7" name="CompanyCountry" ... </select>

<option disabled label="Country/Region" ... </option>

<option value="US">United States</option>

<option value="AF">Afghanistan</option>

<option value="AL">Albania</option>

<option value="DZ">Algeria</option>

<option value="AS">American Samoa</option>

<option value="AD">Andorra</option>

<option value="AI">Anguilla</option>

... </select>

div.country\_field.selectFieldInput div.field.valid select#CompanyCountry-Kmm7 option

select#CompanyCountry-Kmm7>option

Styles Computed Layout Event Listeners DOM Breakpoints Properties

Filter

element.style {

ENG IN 10:59 03-04-2025

## Child or Subchild : Descendant Combinator ( ) is used to select child or Subchild

Syntax : tagName[AttributeName='AttributeValue'] tagName[AttributeName='AttributeValue']

Sign Up Log In

Tutorials Exercises Services

HTML CSS JAVASCRIPT SQL PYTHON JAVA PHP HOW TO W3.C

Demo: Dropdown Examples

Move the mouse over the examples below:

Dropdown Text

Dropdown Menu

Other:

Basic Dropdown

Create a dropdown box that appears when the user moves the mouse over an element.

Example

<style>

30°C Sunny

Google Chrome isn't your default browser Set as default

Elements Console Sources Network

div.w3-third > div.dropdown.dropdown2 > span.dropdown > Dropdown Text

div.w3-third > div.dropdown.dropdown2 > div.dropdown-content > a href="javascript:void(0)">Link 1</a>

div.w3-third > div.dropdown.dropdown2 > div.dropdown-content > a href="javascript:void(0)">Link 2</a>

div.w3-third > div.dropdown.dropdown2 > div.dropdown-content > a href="javascript:void(0)">Link 3</a>

div.w3-white div#main.w3-col.l10.m12 div.w3-row div.w3-third div.dropdown.dropdown2

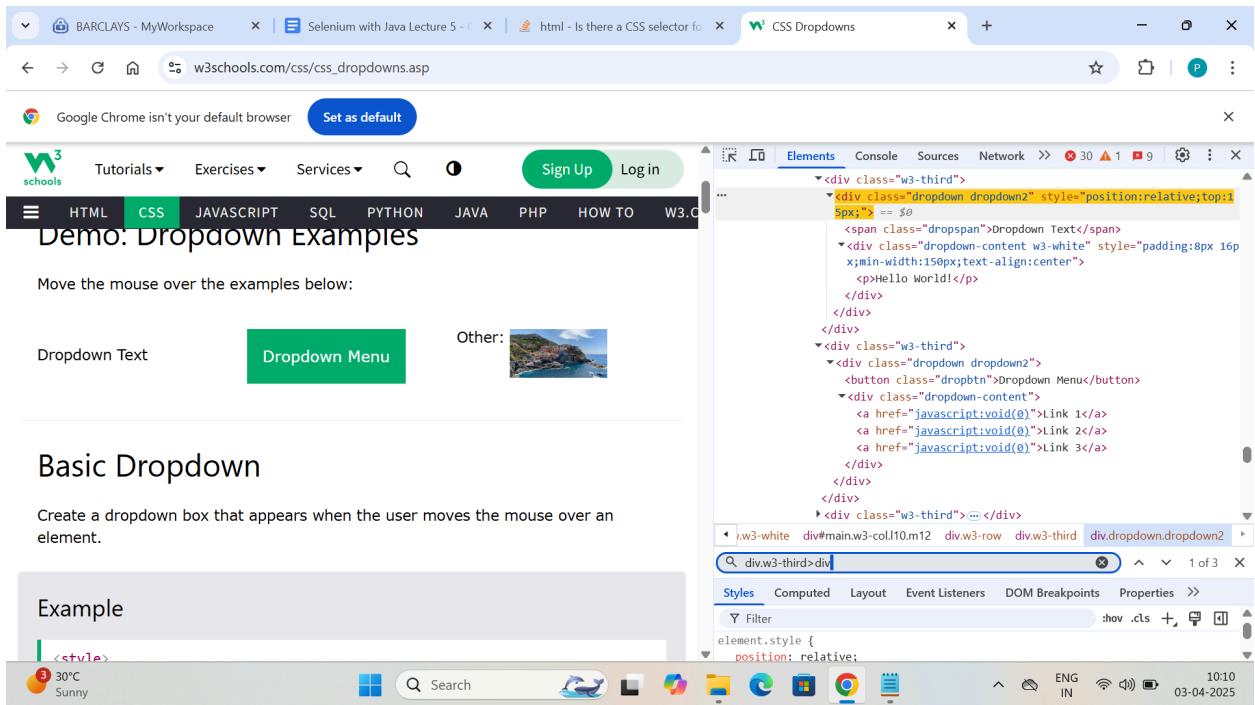
div.w3-third div

Styles Computed Layout Event Listeners DOM Breakpoints Properties

Filter

element.style { position: relative;

ENG IN 10:10 03-04-2025



## Finding Next Sibling Elements:

- Sibling elements are located using the + operator
- Adjacent sibling combinator(+) separates two css selectors and matches the second webelement only if it immediately follows the first webelement, and both are the child of the same parent webelement.

Syntax : tagName[AttributeName='AttributeValue']+tagName[AttributeName='AttributeValue']

The screenshot shows a Salesforce developer sign-up form in a browser window. The form includes fields for First name, Last name, Job title, Work email, Company, and Country/Region (set to India). There are also terms of service and reCAPTCHA checkboxes. The Google Chrome DevTools Elements tab is open, displaying the DOM structure. A specific element, `div.field-container-50+div`, is selected in the tree view. The bottom status bar shows the URL `salesforce.com/form/developer-signup/?d=pb`.

Example: `div.field-container-50+div`

## first-child, last-child, nth-child, nth-last-child :

**first-child** : returns the first element from the group of sibling elements.

A free Salesforce Platform environment

First name

Job title

Company  India

Your org may be provisioned on or migrated to Hyperforce, Salesforce's public cloud infrastructure.

I agree to the Main Services Agreement – Developer Services and Salesforce Program Agreement. I acknowledge, as described in the Developer Documentation: (1) the Developer Edition includes autonomous and other generative AI features; and (2) Salesforce may limit use of those features and the org, and may terminate any org that has been inactive for 45 days.

We value your privacy. To learn more, visit our [Privacy Statement](#).

I'm not a robot

30°C Sunny

Google Chrome isn't your default browser Set as

Developer Edition with A

Elements Console Sources Network

```
<select id="CompanyCountry-zuBe" name="Country/Region" aria-required="true" required="true">
    <option disabled="" label="Country/Region">Country/Regions</option>
    <option value="US">United States</option>
    <option value="AF">Afghanistan</option>
    <option value="AL">Albania</option>
    <option value="DZ">Algeria</option>
    <option value="AS">American Samoa</option>
    <option value="AD">Andorra</option>
    <option value="AI">Anguilla</option>
    <option value="AQ">Antarctica</option>
    <option value="AG">Antigua And Barbuda</option>
    <option value="AR">Argentina</option>
    <option value="AM">Armenia</option>
    <option value="AW">Aruba</option>
    <option value="AU">Australia</option>
```

Styles Computed Layout Event Listeners DOM Breakpoints Properties

element.style { }

10:53 03-04-2025

Example: `select[name='CompanyCountry']>:first-child`

**last-child** : returns the last element from the group of sibling elements.

A free Salesforce Platform environment

First name

Job title

Company  India

Your org may be provisioned on or migrated to Hyperforce, Salesforce's public cloud infrastructure.

I agree to the Main Services Agreement – Developer Services and Salesforce Program Agreement. I acknowledge, as described in the Developer Documentation: (1) the Developer Edition includes autonomous and other generative AI features; and (2) Salesforce may limit use of those features and the org, and may terminate any org that has been inactive for 45 days.

We value your privacy. To learn more, visit our [Privacy Statement](#).

I'm not a robot

30°C Sunny

Google Chrome isn't your default browser Set as

Developer Edition with A

Elements Console Sources Network

```
<select id="CompanyCountry-zuBe" name="Country/Region" aria-required="true" required="true">
    <option value="US">United States</option>
    <option value="AF">Afghanistan</option>
    <option value="AL">Albania</option>
    <option value="DZ">Algeria</option>
    <option value="AS">American Samoa</option>
    <option value="AD">Andorra</option>
    <option value="AI">Anguilla</option>
    <option value="AQ">Antarctica</option>
    <option value="AG">Antigua And Barbuda</option>
    <option value="AR">Argentina</option>
    <option value="AM">Armenia</option>
    <option value="AW">Aruba</option>
    <option value="AU">Australia</option>
    <option value="EH">Western Sahara</option>
    <option value="YE">Yemen</option>
    <option value="ZW">Zambia</option>
    <option value="ZW">Zimbabwe</option>
```

Styles Computed Layout Event Listeners DOM Breakpoints Properties

element.style { }

10:54 03-04-2025

Example: `select[name='CompanyCountry']>:last-child`

**nth-child** : returns elements based on their position in the group of siblings.

A free Salesforce Platform environment

First name

Job title

Company

Your org may be provisioned on or migrated to Hyperforce, Salesforce's public cloud infrastructure.

I agree to the [Main Services Agreement – Developer Services and Salesforce Program Agreement](#). I acknowledge, as described in the Developer Documentation: (1) the Developer Edition includes autonomous and other generative AI features; and (2) Salesforce may limit use of those features and the org, and may terminate any org that has been inactive for 45 days.

We value your privacy. To learn more, visit our [Privacy Statement](#).

I'm not a robot 

30°C Sunny

Google Chrome isn't your default browser [Set as](#)

Country/Region

- United States
- Afghanistan
- Albania
- Algeria
- American Samoa
- Andorra
- Anguilla
- Antarctica
- Antigua And Barbuda
- Argentina
- Armenia
- Aruba
- Australia
- India

Elements Console Sources Network ...

```
option value="AF">>Afghanistan</option>
== $0
option value="AL">>Albania</option>
option value="DZ">>Algeria</option>
option value="AS">>American Samoa</option>
[el slot]
<option value="AD">>Andorra</option>
option value="AI">>Anguilla</option>
option value="AQ">>Antarctica</option>
option value="AG">>Antigua And Barbuda
[el slot]
<option value="AR">>Argentina</option>
option value="AM">>Armenia</option>
option value="AW">>Aruba</option>
[el slot]
option value="AU">>Australia</option>
option value="AT">>Austria</option>
option value="AZ">>Azerbaijan</option>
option value="BS">>Bahamas, The</option>
option value="BH">>Bahrain</option>
[el slot]
option value="BD">>Bangladesh</option>
```

Styles Computed Layout Event Listeners DOM Breakpoints Properties >

Filter element.style {

hov .cls +

ENG IN 03-04-2025

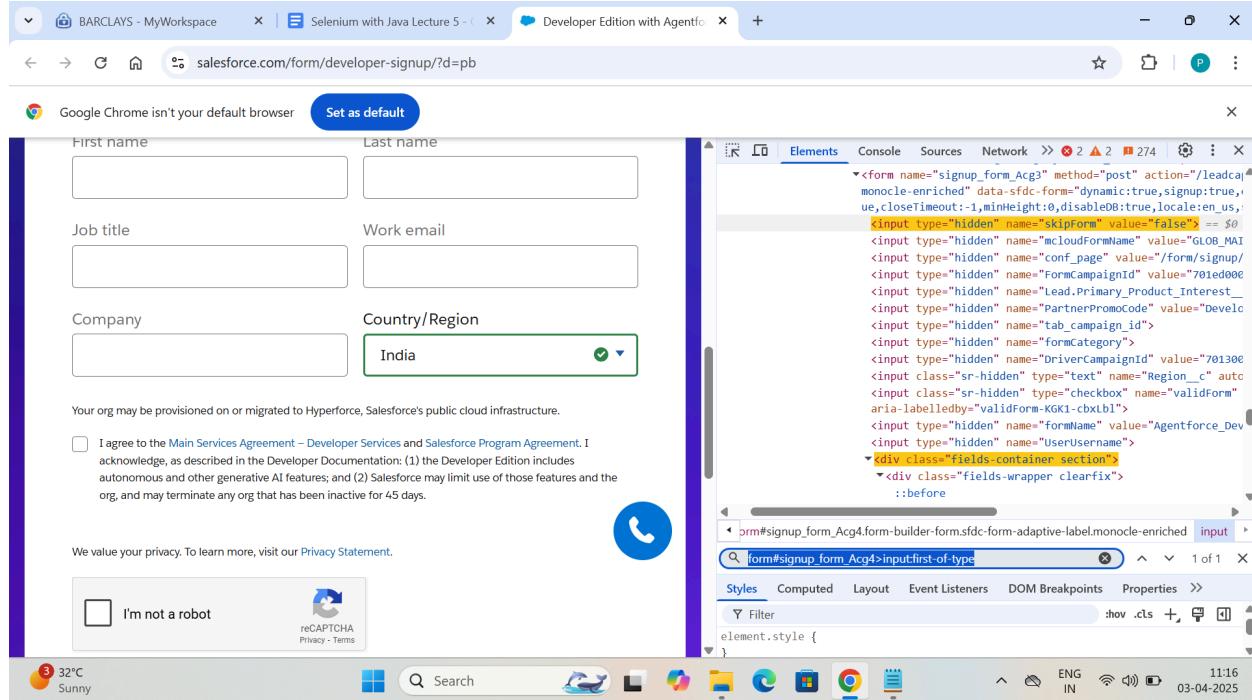
Example: `select[name='CompanyCountry']>:nth-child(3)`

**nth-last-child** : returns elements based on their position among a group of siblings, counting from the end.

Example: `select[name='CompanyCountry']>:nth-last-child(3)`

# first-of-type, last-of-type, nth-of-type():

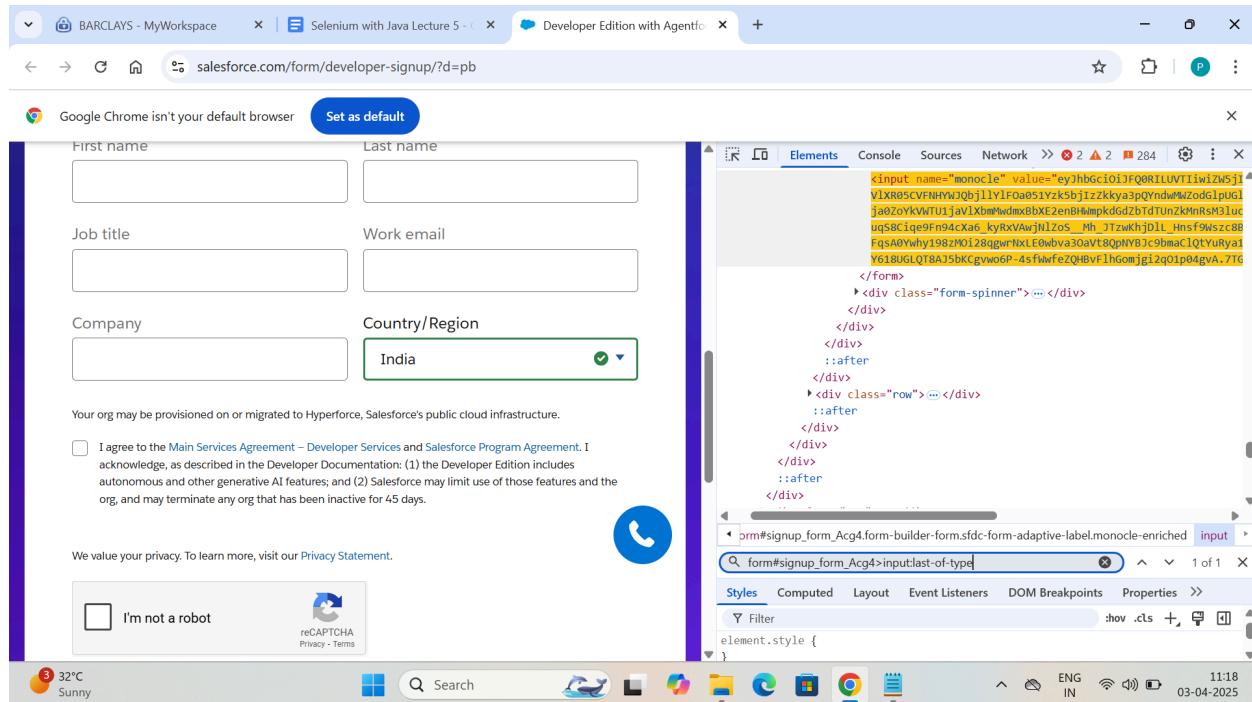
**First-of-type:** returns the first element of its type among a group of sibling elements



The screenshot shows a Salesforce developer sign-up form. On the right, the browser's developer tools are open with the 'Elements' tab selected. A search bar at the top of the tools panel contains the query 'form#signup\_form\_Acg4>input:first-of-type'. The results list shows one element: `<input type="hidden" name="skipForm" value="false">`. This element is highlighted in blue in both the DOM tree and the preview pane below. The preview pane shows a small snippet of the page code with the highlighted element.

Example: `form#signup_form_Acg4>input:first-of-type`

**last-of-type:** returns the last element of its type among a group of sibling elements



The screenshot shows the same Salesforce developer sign-up form as the previous example, but with the developer tools showing the 'last-of-type' selector. The search bar at the top of the tools panel now contains the query 'form#signup\_form\_Acg4>input:last-of-type'. The results list shows one element: `<input type="hidden" name="monocle" value="eyjhbgc101JFQ0R1LWYT1wizW5j1V1Xh05CVHnWYqjb1lyIrcou051y1zk5bjizzkkyaaqpfrdw=...zcodG1pu61ja0zoyKw1t01av1x0mNmmdxBXe2enBfampkdg2zb1d1nzkhnMs3lucuqSRC1qe9f94cxA6_kyRxVwjlNz05_Mh_37zwkhj01_lnsf9wsC8BFGsAv9hyh198x0128qgeNxL0wbv3a0vt8qNVy3c9hmaC1qtYuTyaiY618UGLQ78AJ5bkCgvu06P-4sfWweZQHBF1hcomjg12q0p04gvA..7T6" />`. This element is highlighted in blue in the DOM tree and preview pane. The preview pane shows a long string of encoded values for the monocle input field.

Example: `form#signup_form_Acg4>input:last-of-type`

**nth-of-type()**: matches elements of a given type, based on their position among a group of sibling elements

The screenshot shows a Salesforce developer sign-up form in a browser window. The developer tools are open, specifically the Elements tab, which displays the DOM structure of the page. A search bar in the tools is set to 'form#signup\_form\_Acg4>input:last-of-type'. The results show several input fields, with one specific input field highlighted in green, indicating it is the last-of-type element matching the selector. The rest of the page content includes fields for First name, Last name, Job title, Work email, Company, and Country/Region (set to India), along with a reCAPTCHA and privacy-related checkboxes.

Example: `form#signup_form_Acg4>input:nth-of-type(3)`

