

# Playwright

XPATH

How? When?

# The Golden Rule

What

What is Xpath?  
XML path –address of the element in DOM

When

**When to use?**  
Basic locators are not available / changing dynamically.

How

How to write xpath?  
Based on the element path and syntax

# Types of Xpath

## ➤ Absolute Xpath



- ☐ You must follow the **exact route** and **fixed stops** of the railway line to reach your destination

## ➤ Relative Xpath



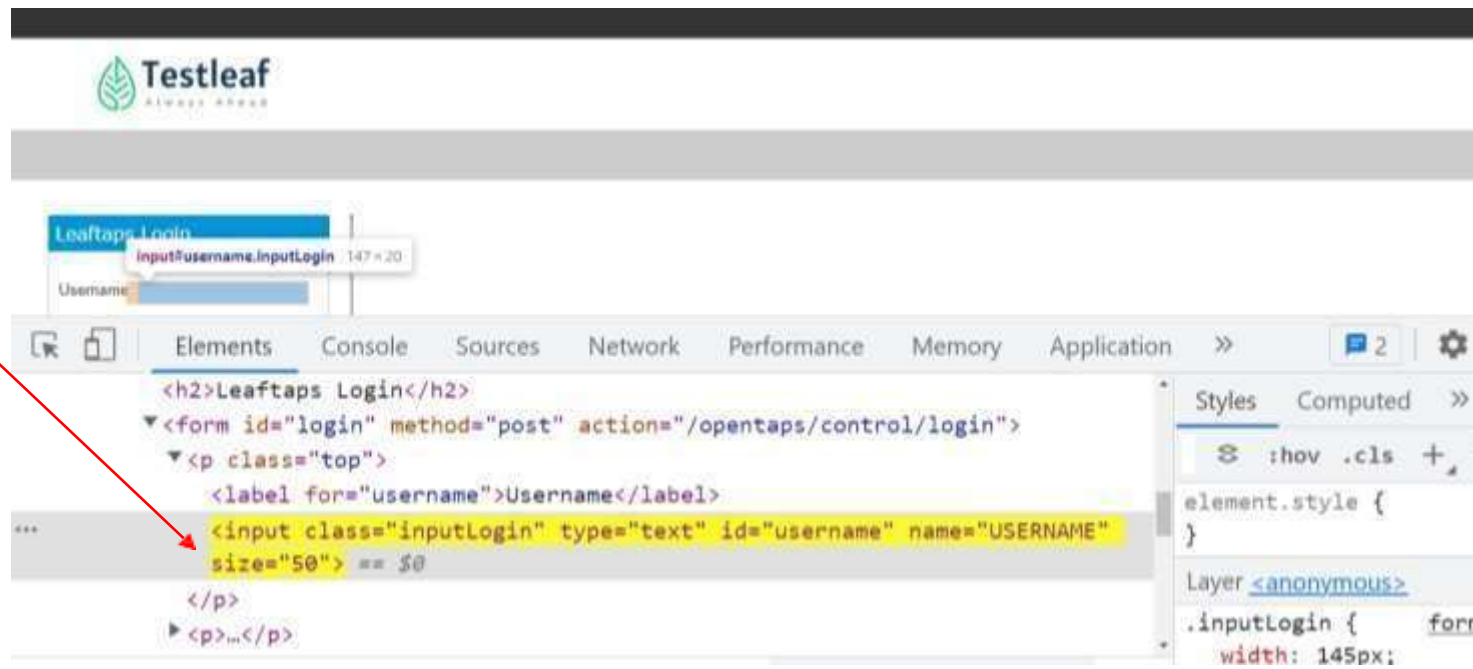
- ☐ You can take **shortcuts** or **adjust your route** based on current traffic or road conditions.

# Absolute Xpath

- Long path, Starts from root tag(<html>) and ends at the target element tag.
- Specifies the full hierarchy to target an element.
- Prone to breaking when the page structure changes.
- Starts with “/” and traverse through each and every tag in hierarchy to reach the required WebElement

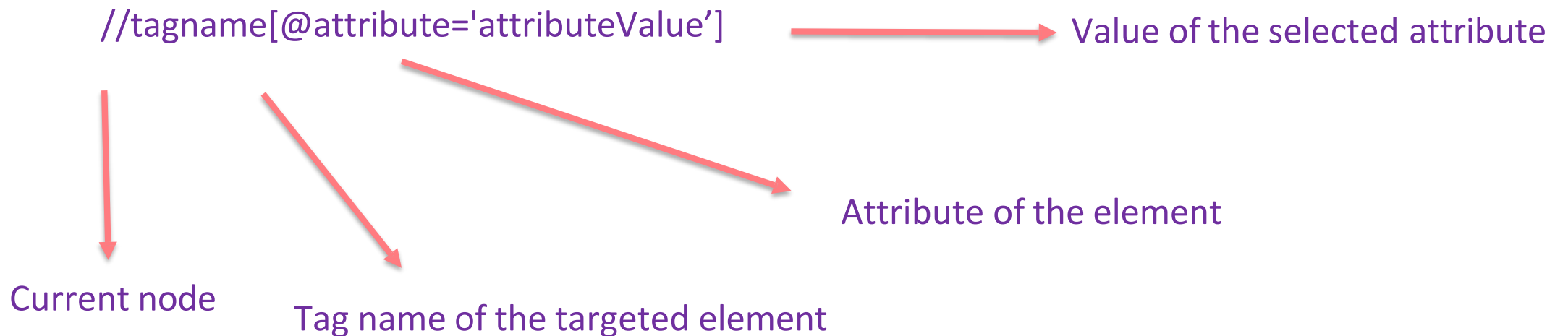
**The syntax for the absolute xpath is**  
**/html/body/div[2]/div/div/form/p/input**

-Rarely used because of unreliable



# Relative XPath

- Begins from the current element, not necessarily the root. Defines a
  - shorter, context-based path to locate an element
  - Offers more robust and flexible element targeting, suitable for dynamic web pages.
- Relative XPath starts with the “//” which locates the inner or middle part of the DOM.
- General syntax of the Relative XPath is:



# Types of Relative Locators

## Relative Locators

- Basic XPath
- Advanced XPath

The screenshot displays the Testleaf login page. The developer tools are open, showing the 'Elements' panel with the following HTML structure:

```
<h2>Leaftaps Login</h2>
<form id="login" method="post" action="/opentaps/control/login">
  <p class="top">
    <label for="username">Username</label>
    <input class="inputLogin" type="text" id="username" name="USERNAME" size="50">
  </p>
  <p>...</p>
  <p>...</p>
</form>
```

The 'Styles' panel shows the following CSS for the selected input element:

```
.inputLogin {
  width: 145px;
  background-color: #FAFAFA;
  border: 1px solid #ccc;
}
```

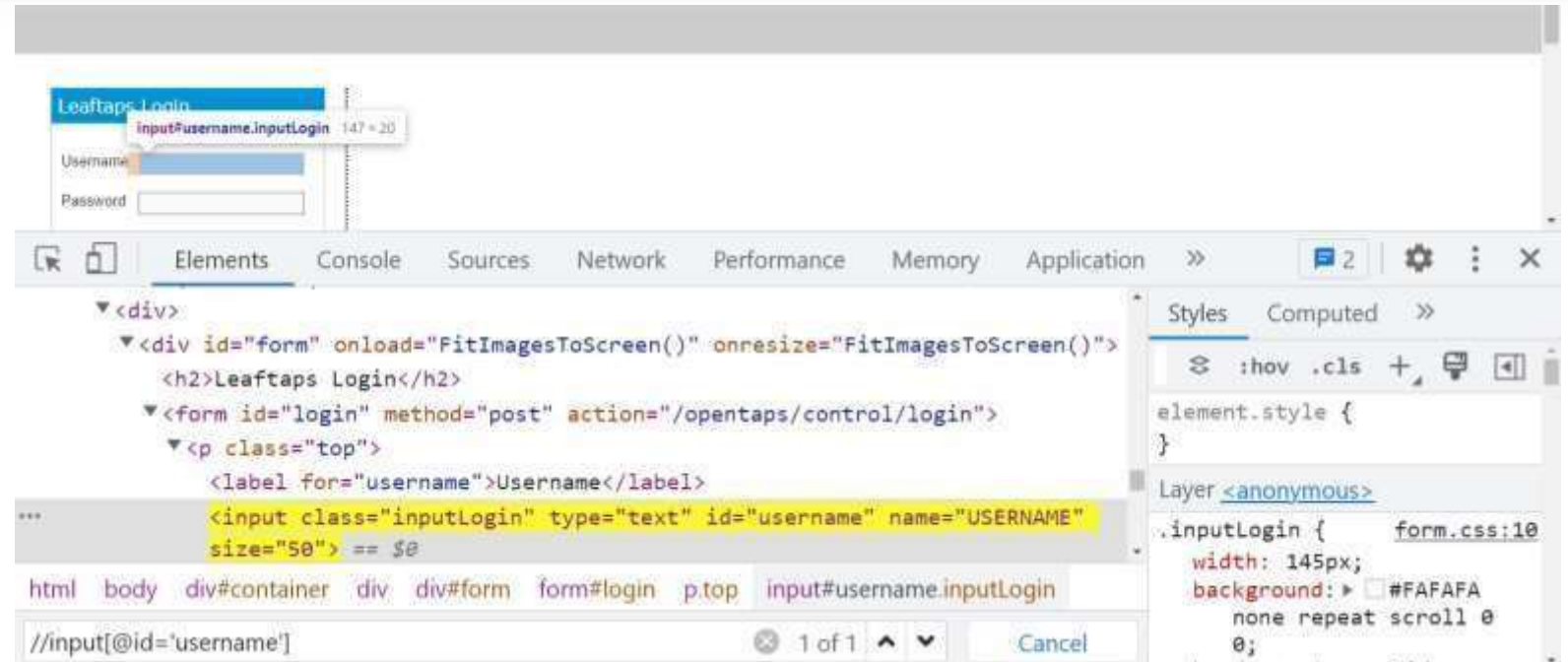
Annotations in the image point to specific parts of the HTML code:

- Text**: Points to the text "Username" inside the label element.
- Attribute**: Points to the `class` attribute of the input element.
- Attribute Value**: Points to the value `inputLogin` of the `class` attribute.

# Basic Xpath – Attribute Based

## Attribute Based XPath:

->uses the attribute and attribute value of the WebElement in the DOM.



Syntax:

**`//tagName[@attribute='attributeValue']`**

Example

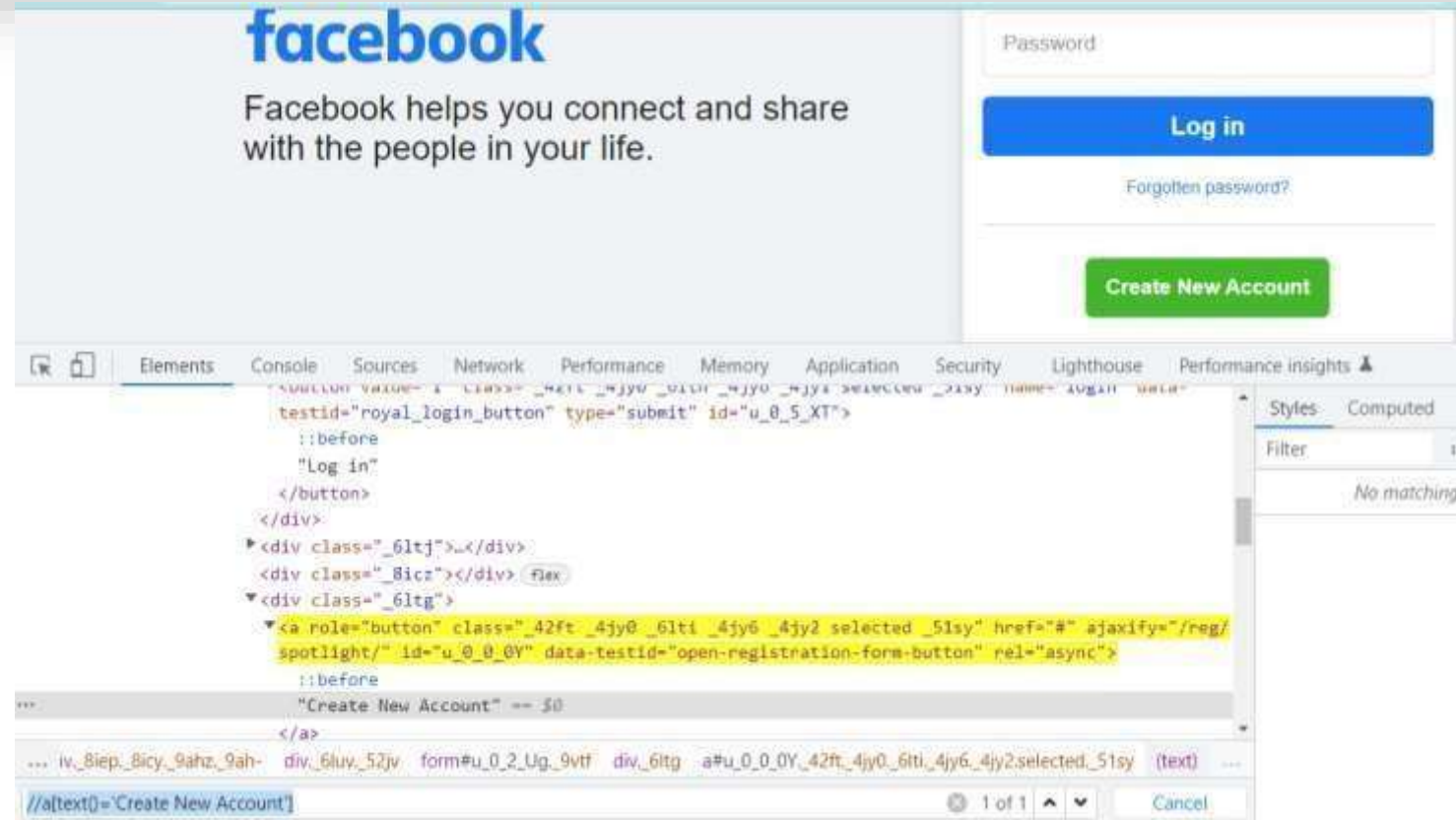
**`//input[@id='username']`**

**Contd..**

# Basic Xpath – Text Based

## Text Based Xpath:

->based on the text displayed for the WebElement in dom.



## Syntax:

`//tagName[text()='text value in DOM']`

## Example:

`//a[text()='Create New Account']`

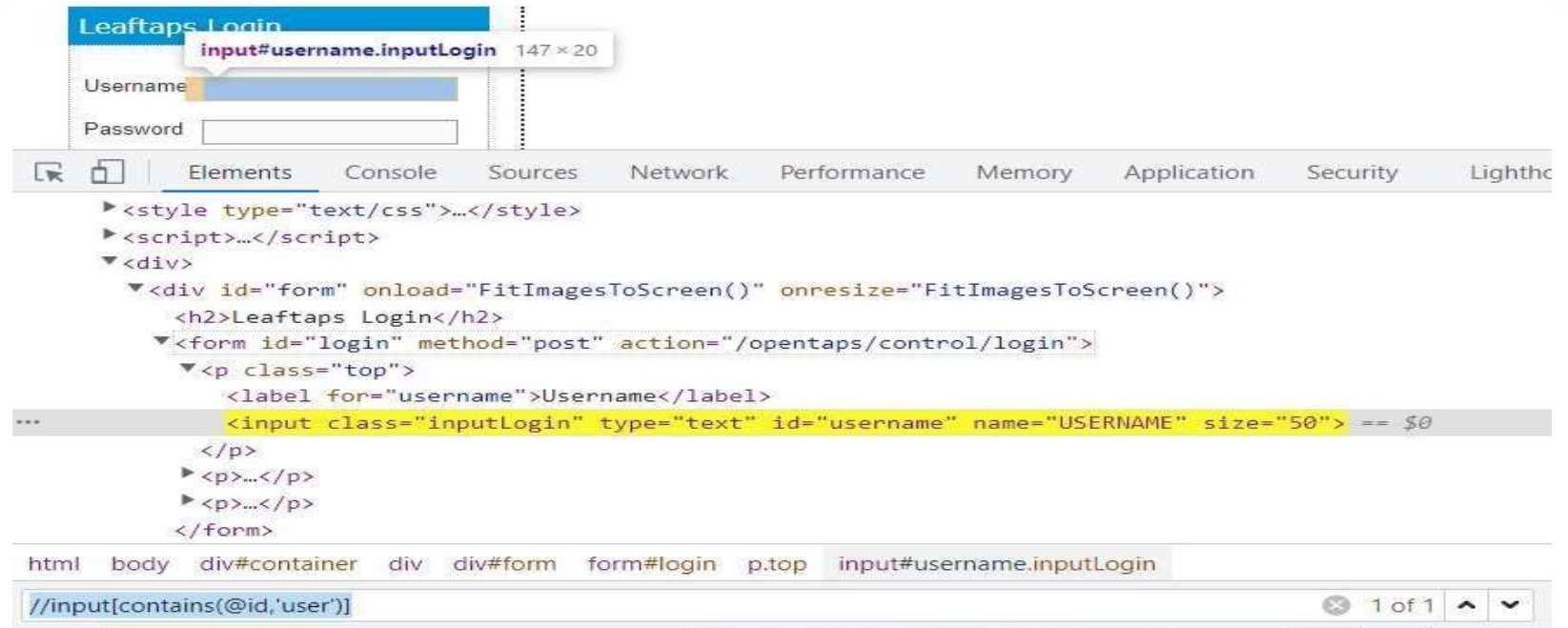
Contd..



# Basic Xpath – Partial Match

**Partial attribute Based Xpath:**  
=>uses the partial attribute value  
of the WebElement

**\*Contains ->check for the given  
attribute  
value with the value in DOM**



**Syntax:**

`//tagName[contains(@attribute, 'Partial value of the attribute')]`

**Example:**

`//input[contains(@id,'user')]`

**Contd..**

# Basic Xpath – Partial Match

## Partial Text Based Xpath:

->uses the partial text value of the WebElement



Syntax:

`//tagname[contains(text(),' Partial text in my DOM' )]`

Example:

`//a[contains(text(),'Create')]`

Contd..

# Basic Xpath - Collection

## Collection Based Xpath:

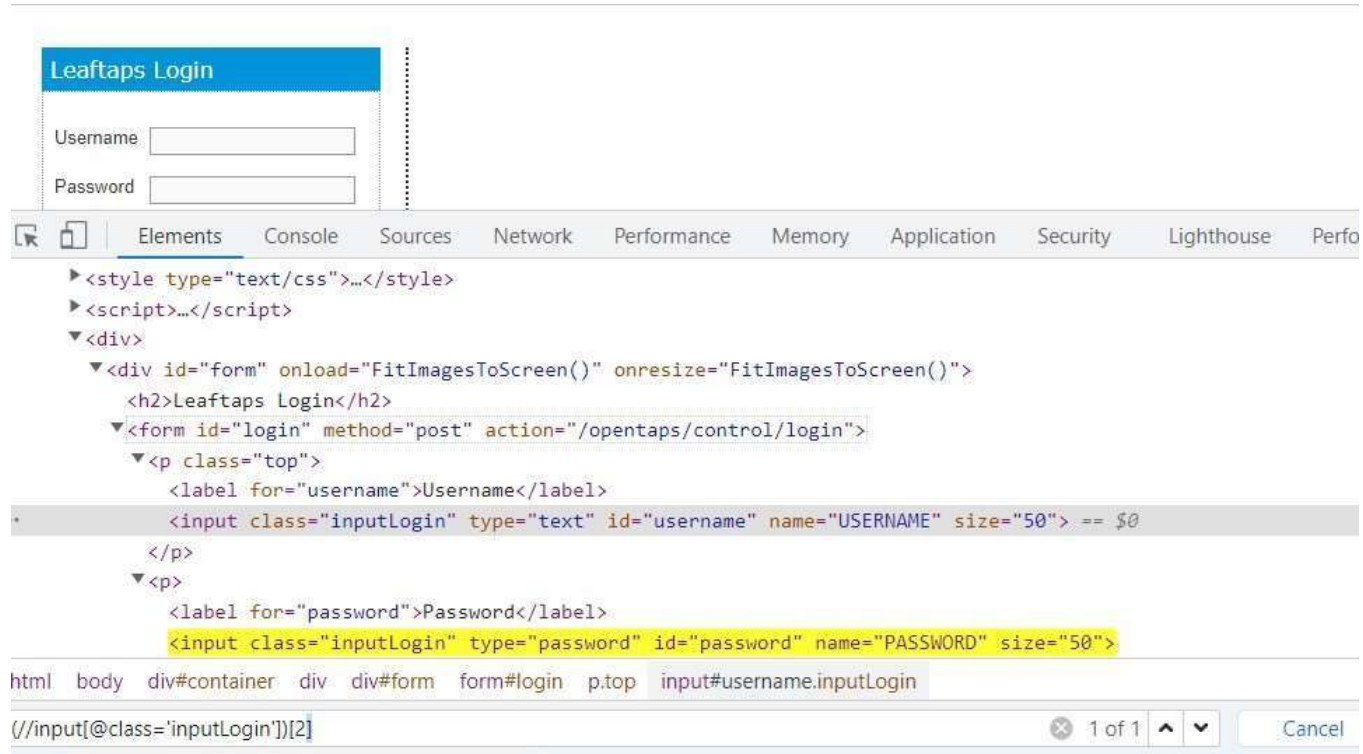
-> when there is multiple

matches of found for a xpath

->uses index to match the exact

element

->Index starts with 1



Syntax:

`(//tagName[@attribute='attributeValue'])[index]`

Example

`(//input[@class='inputLogin'])[2]`

# Summary

- **Basic Xpath-What , when, How?**
- **2 types- Absolute and Relative xpath**
- **Most recommended –Relative xpath**

# Let's learn few actions & add a new Test

Practice a simple program by creating a lead in the Leaftap application and understand few actions:

- Typing
- Clicking
- Navigating