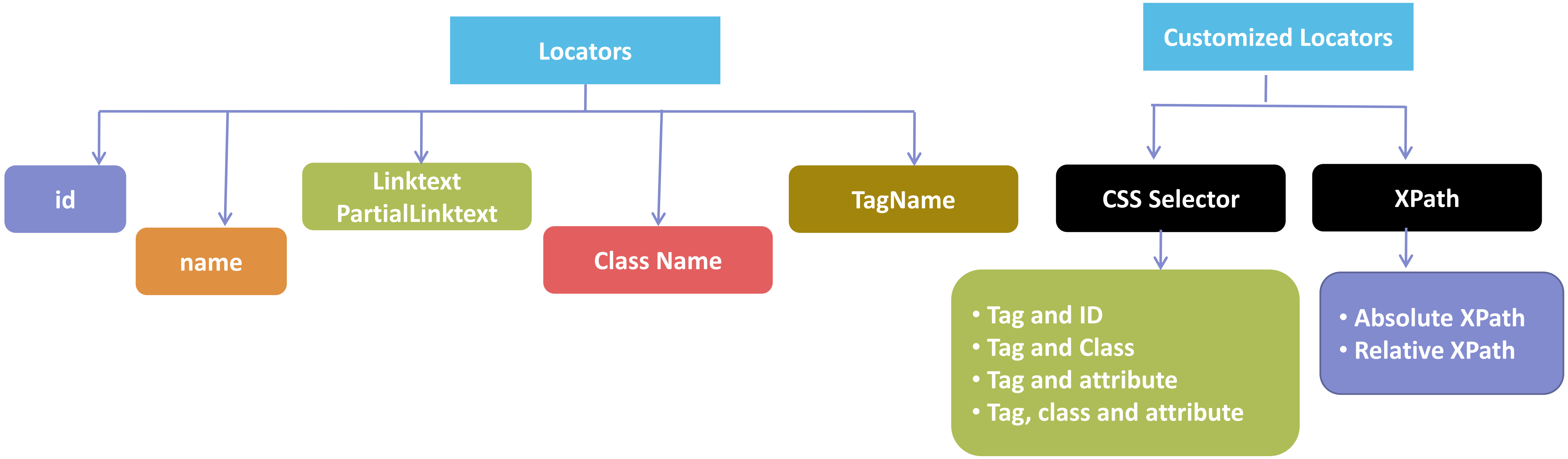
Selenium Locators

Types of Locators

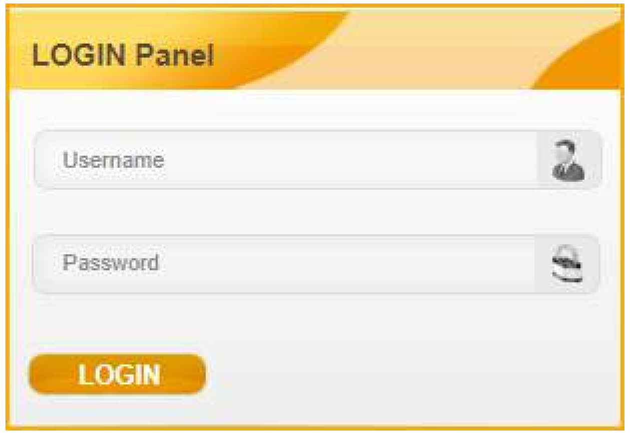
- We can identify various elements on the web using **Locators**.
- Locators are addresses that identify a web element uniquely within the page.

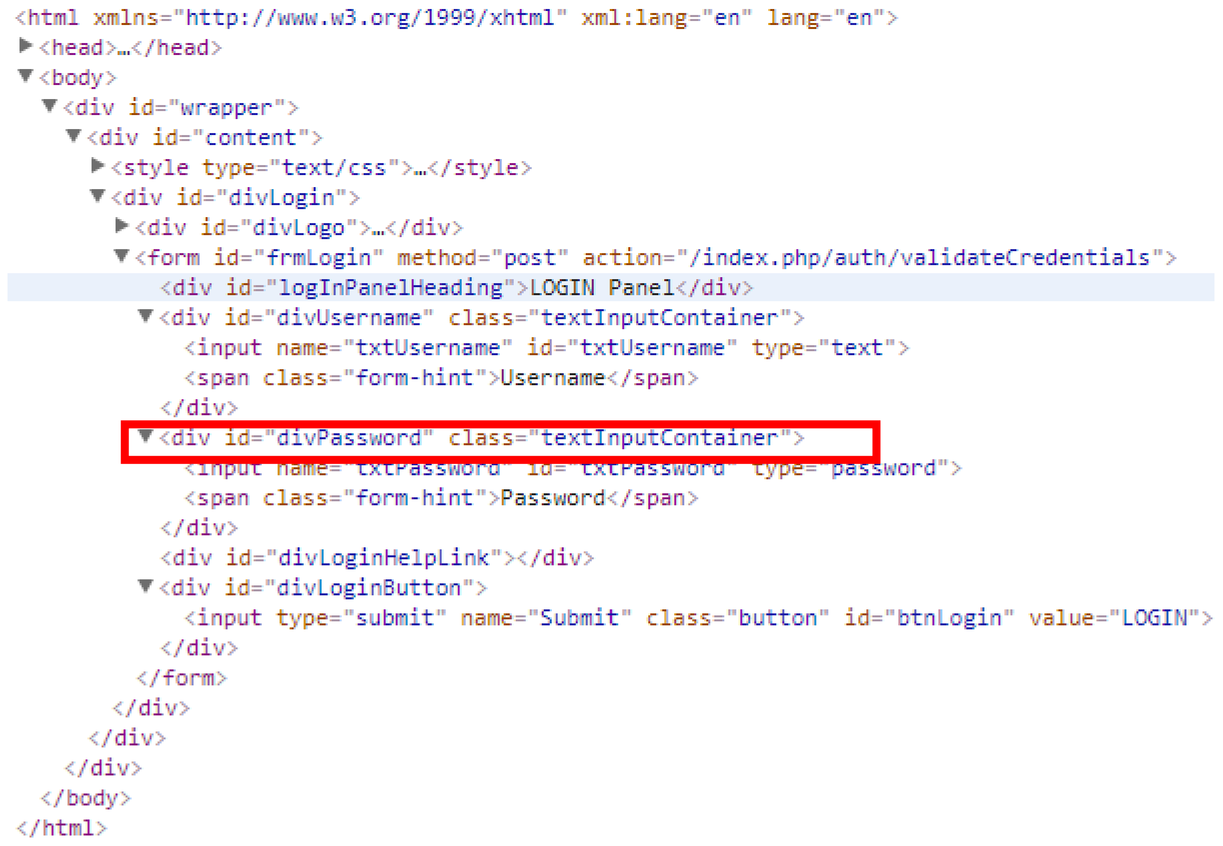


Locators

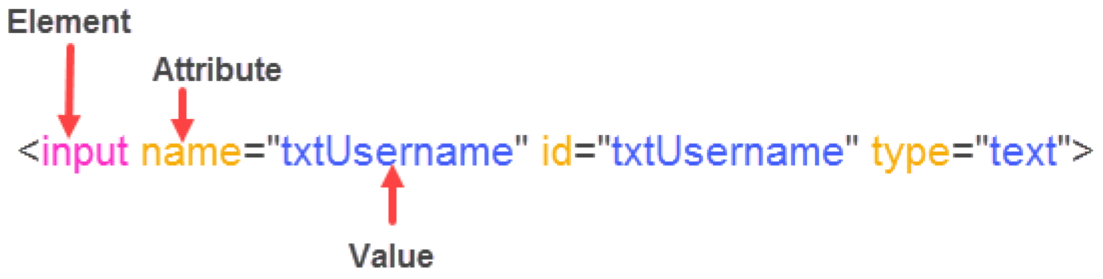
- id
- name
- linkText
- Partial LinkText
- class
- TagName

HTML Structure





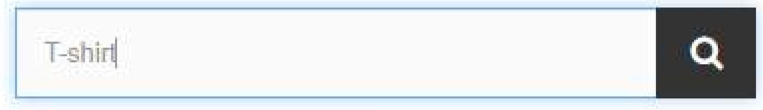
```
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">
▶<head>…</head>
▼<body>
  ▼<div id="wrapper">
    ▼<div id="content">
      ▶<style type="text/css">…</style>
      ▼<div id="divLogin">
        ▶<div id="divLogo">…</div>
        ▼<form id="frmLogin" method="post" action="/index.php/auth/validateCredentials">
            <div id="logInPanelHeading">LOGIN Panel</div>
          ▼<div id="divUsername" class="textInputContainer">
              <input name="txtUsername" id="txtUsername" type="text">
              <span class="form-hint">Username</span>
            </div>
          ▼<div id="divPassword" class="textInputContainer">
              <input name="txtPassword" id="txtPassword" type="password">
              <span class="form-hint">Password</span>
            </div>
            <div id="divLoginHelpLink"></div>
          ▼<div id="divLoginButton">
              <input type="submit" name="Submit" class="button" id="btnLogin" value="LOGIN">
            </div>
          </form>
        </div>
      </div>
    </div>
  </body>
</html>
```

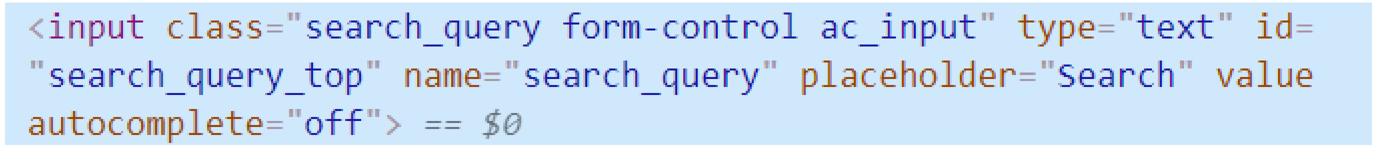


```
<input name="txtUsername" id="txtUsername" type="text">
```

ID

http://automationpractice.com/index.php



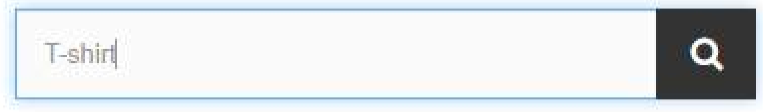


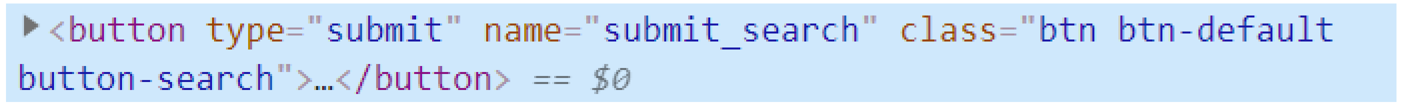
```
<input class="search_query form-control ac_input" type="text" id=
"search_query_top" name="search_query" placeholder="Search" value
autocomplete="off"> == $0
```

```
driver.findElement(By.id("search_query_top")).sendKeys("T-shirt");
```

Name

http://automationpractice.com/index.php

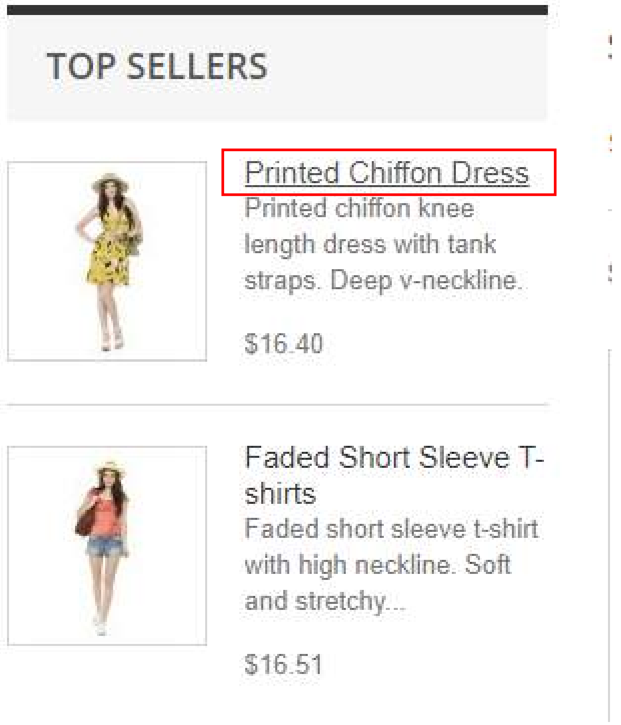




```
▶<button type="submit" name="submit_search" class="btn btn-default button-search">…</button> == $0
```

```
driver.findElement(By.name("submit_search")).click();
```

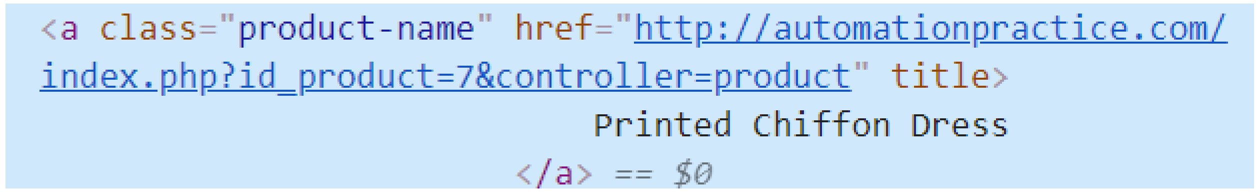
Link Text / Partial Link Text



TOP SELLERS

Printed Chiffon Dress
Printed chiffon knee length dress with tank straps. Deep v-neckline.
$16.40

Faded Short Sleeve T-shirts
Faded short sleeve t-shirt with high neckline. Soft and stretchy...
$16.51

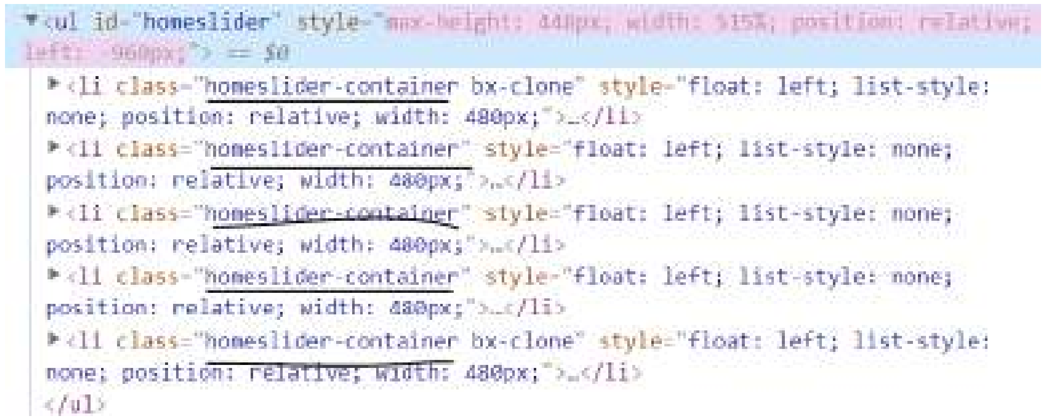


```
<a class="product-name" href="http://automationpractice.com/index.php?id_product=7&controller=product" title>
                    Printed Chiffon Dress
                </a> == $0
```

```
driver.findElement(By.linkText("Printed Chiffon Dress")).click();
driver.findElement(By.partialLinkText("Chiffon Dress")).click();
```

Class Name

http://automationpractice.com/index.php

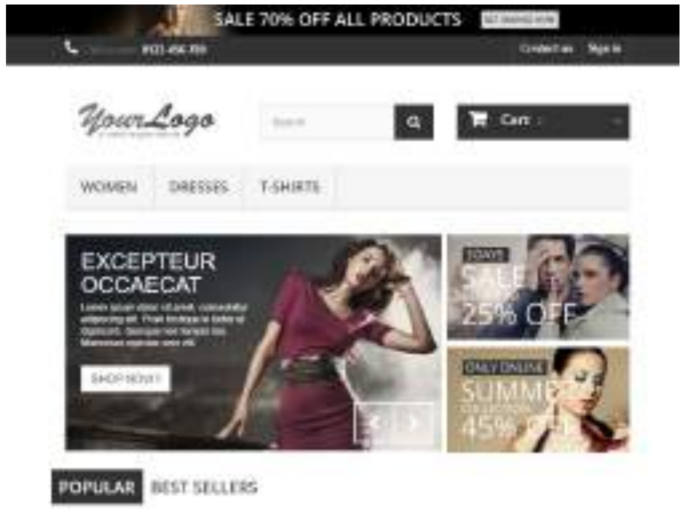


```
▼<ul id="homeslider" style="max-height: 448px; width: 515%; position: relative;
left: -960px;"> == $0
  ▶<li class="homeslider-container bx-clone" style="float: left; list-style:
  none; position: relative; width: 480px;">…</li>
  ▶<li class="homeslider-container" style="float: left; list-style: none;
  position: relative; width: 480px;">…</li>
  ▶<li class="homeslider-container" style="float: left; list-style: none;
  position: relative; width: 480px;">…</li>
  ▶<li class="homeslider-container" style="float: left; list-style: none;
  position: relative; width: 480px;">…</li>
  ▶<li class="homeslider-container bx-clone" style="float: left; list-style:
  none; position: relative; width: 480px;">…</li>
</ul>
```

```java
int sliders=driver.findElements(By.className("homeslider-container")).size();
System.out.println(sliders);
```

TagName

http://automationpractice.com/index.php



```java
int links=driver.findElements(By.tagName("a")).size();
System.out.println(links);
```

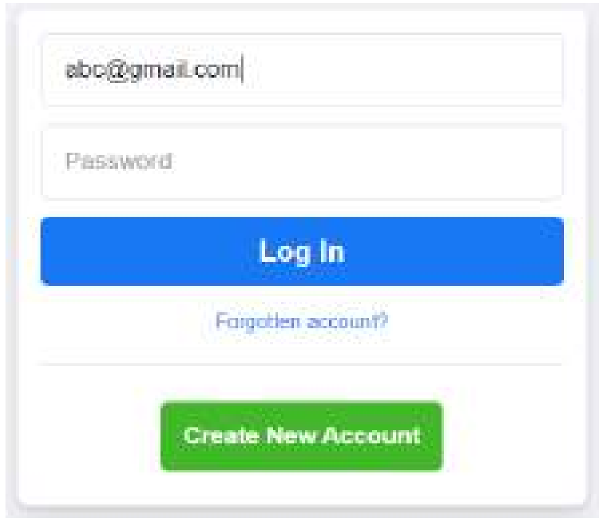
CSS Selectors

CSS Selector - Cascading Style Sheets

- Tag & ID   (OR) #id
- Tag & class   (OR) .class
- Tag & attribute   (OR) [attribute=value]
- Tag , class & attribute

CSS Selector – *Tag* and *ID*

https://www.facebook.com/



```
<input type="text" class="inputtext _55r1 _6luy" name="email" id=
"email" data-testid="royal_email" placeholder="Email address or phone
number" autofocus="1" aria-label="Email address or phone number"> == $0
```

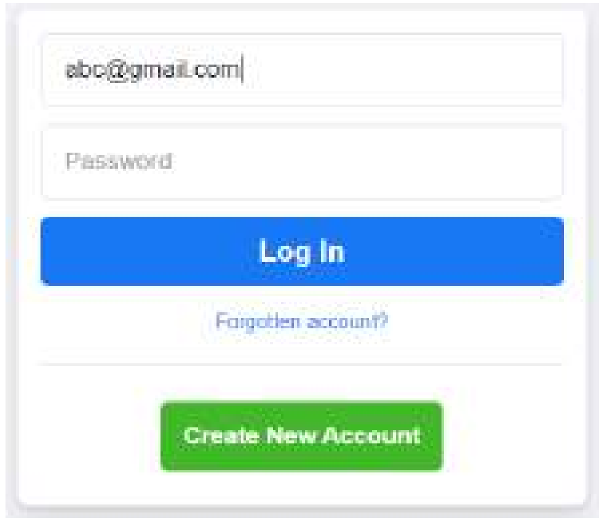
```
driver.findElement(By.cssSelector("#email")).sendKeys("abc@gmail.com");
```

(or)

```
driver.findElement(By.cssSelector("input#email")).sendKeys("abc@gmail.com");
```

CSS Selector – *Tag* and Class

https://www.facebook.com/



```
<input type="text" class="inputtext _55r1 _6luy" name="email" id=
"email" data-testid="royal_email" placeholder="Email address or phone
number" autofocus="1" aria-label="Email address or phone number"> == $0
```

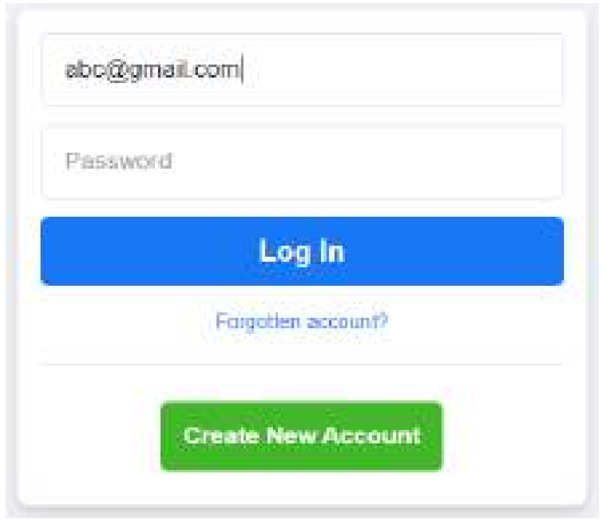
```
driver.findElement(By.cssSelector(".inputtext")).sendKeys("abc@gmail.com");
```

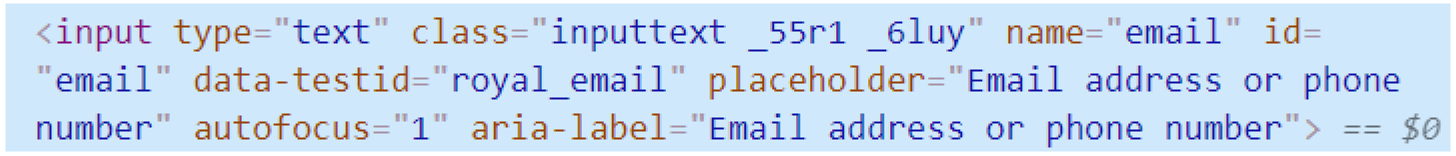
(or)

```
driver.findElement(By.cssSelector("input.inputtext")).sendKeys("abc@gmail.com");
```

CSS Selector – *Tag* and Attribute

https://www.facebook.com/



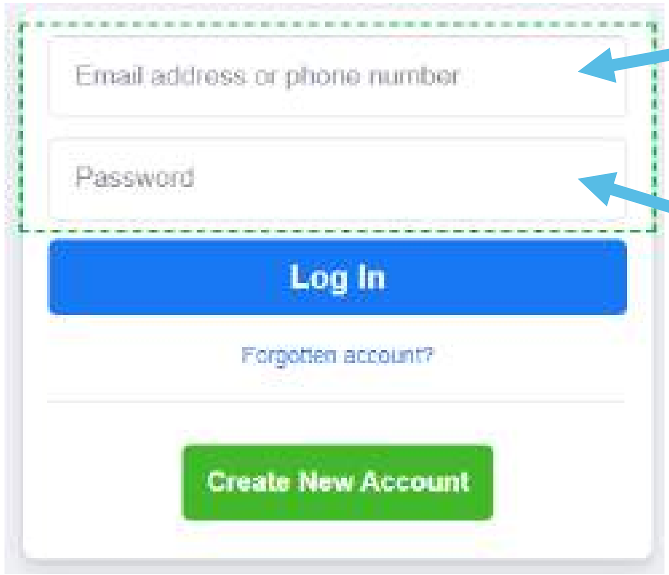


```
<input type="text" class="inputtext _55r1 _6luy" name="email" id=
"email" data-testid="royal_email" placeholder="Email address or phone
number" autofocus="1" aria-label="Email address or phone number"> == $0
```

```
driver.findElement(By.cssSelector("[name=email]")).sendKeys("abc@gmail.com");

                                    (or)

driver.findElement(By.cssSelector("input[name=email]")).sendKeys("abc@gmail.com");
```

CSS Selector - *Tag, class* and *attribute*

https://www.facebook.com/



```
▼<div class="_6lux">
    <input type="text" class="inputtext _55r1 _6luy" name=
    "email" id="email" data-testid="royal_email" placeholder=
    "Email address or phone number" autofocus="1" aria-label=
    "Email address or phone number" style>
  </div>
▼<div class="_6lux">
    <input type="password" class="inputtext _55r1 _6luy" name=
    "pass" id="pass" data-testid="royal_pass" placeholder=
    "Password" aria-label="Password">
  </div>
```

```
driver.findElement(By.cssSelector("input.inputtext[data-testid=royal_email]")).sendKeys("abc@gmail.com"); //Email
driver.findElement(By.cssSelector("input.inputtext[data-testid=royal_pass]")).sendKeys("abc"); //Password
```

XPath

XPath

1. **What Is XPath?**
2. **Types Of XPaths**
   - ➢ Absolute
   - ➢ Relative
3. **How to capture XPath?**
4. **Writing Dynamic XPath by different ways:**
   - ➢ Using ‘OR’ & ‘AND’
   - ➢ Using Contains()
   - ➢ Using Starts-With()
   - ➢ Using Text()
   - ➢ Chained XPath

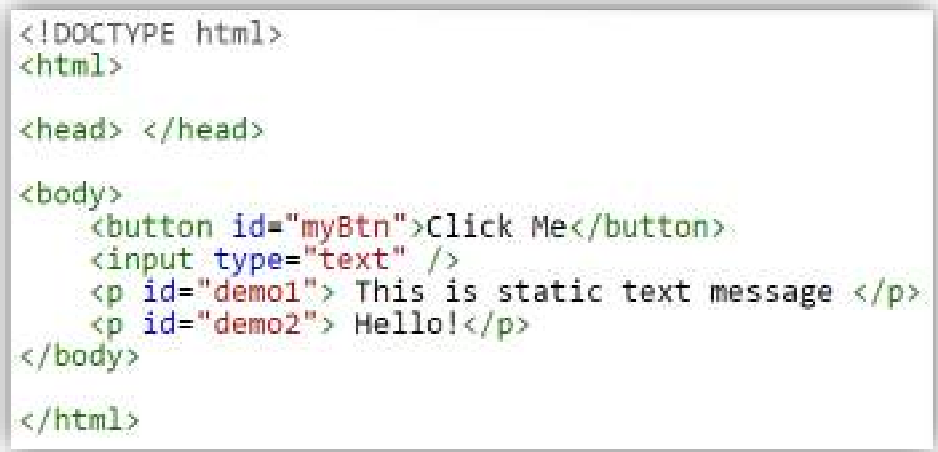
What is XPath?

- XPath is defined as **XML path**.
- **It is a syntax or language for finding any element on the web page using XML path expression**.
- XPath is used to find the location of any element on a webpage using **HTML** DOM structure.
- XPath can be used to navigate through elements and attributes in DOM.

DOM – Document Object Model

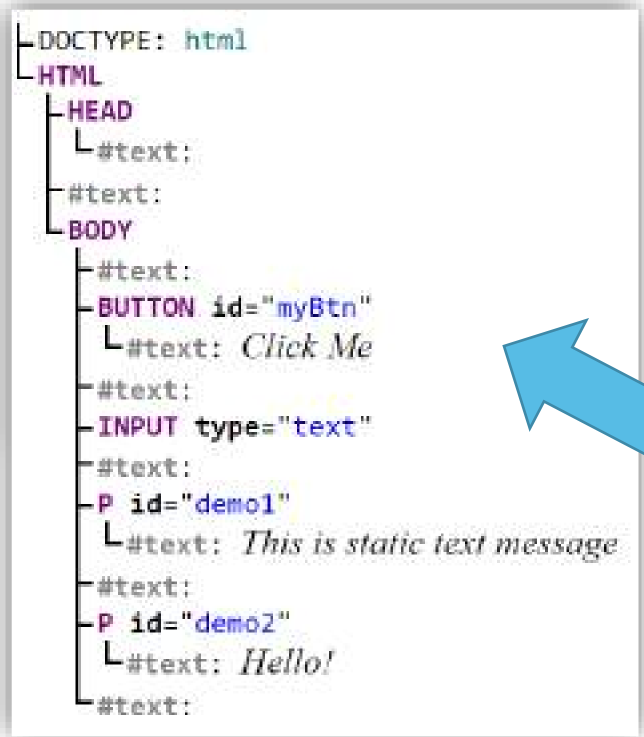
- DOM is an API Interface provided by browser.
- When a web page is loaded, the browser creates a **D**ocument **O**bject **M**odel of the page.

HTML



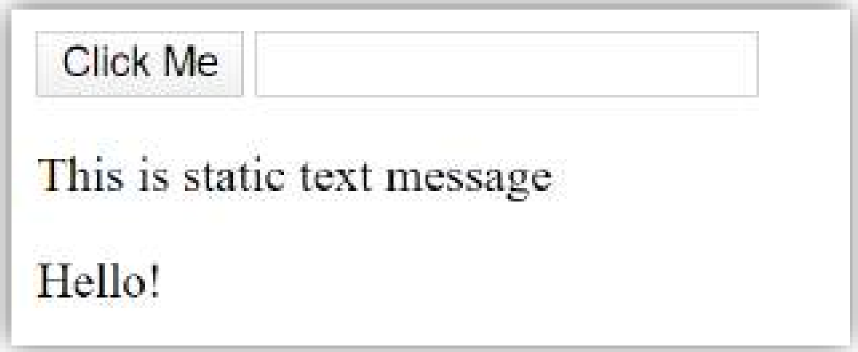
```
<!DOCTYPE html>
<html>

<head> </head>

<body>
    <button id="myBtn">Click Me</button>
    <input type="text" />
    <p id="demo1"> This is static text message </p>
    <p id="demo2"> Hello!</p>
</body>

</html>
```

DOM View



```
└DOCTYPE: html
└HTML
  ├HEAD
  │ └#text:
  ├#text:
  └BODY
    ├#text:
    ├BUTTON id="myBtn"
    │ └#text: Click Me
    ├#text:
    ├INPUT type="text"
    ├#text:
    ├P id="demo1"
    │ └#text: This is static text message
    ├#text:
    ├P id="demo2"
    │ └#text: Hello!
    └#text:
```

Rendered View



This is static text message

Hello!

XPath works here

Absolute XPath

- It is the direct way to find the element.
- The disadvantage of the absolute XPath is that if there are any changes made in the path of the element then that XPath gets failed.
- It begins with the single forward slash(/) ,which means you can select the element from the root node.
- Below is the example of an absolute XPath expression of the element
- Ex:

Absolute Xpath : /html[1]/body[1]/div[1]/div[1]/header[1]/div[3]/div[1]/div[1]/div[1]/a[1]/img[1]

Relative XPath

- Relative XPath the path starts from the middle of the HTML DOM structure.
- It starts with the double forward slash (//), which means it can search the element anywhere at the webpage.
- You can start from the middle of the HTML DOM structure and no need to write long XPath.

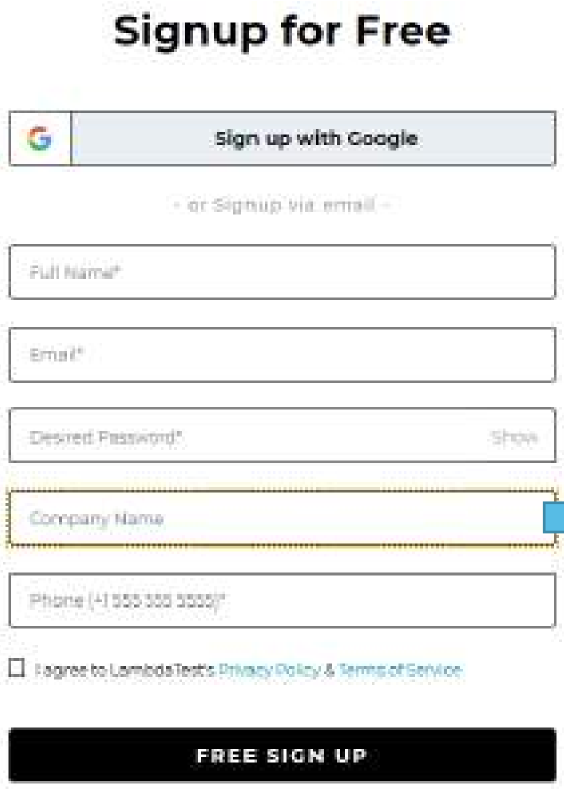
Ex:

Relative Xpath : `//img[@class='logo img-responsive']`

Syntax for Relative XPath

- XPath contains the path of the element situated at the web page. Standard syntax for creating XPath is.
- **// :** Select current node.
- **Tagname:** Tagname of the particular node.
- **@:** Select attribute.
- **Attribute:** Attribute name of the node.
- **Value:** Value of the attribute.
- Xpath=//tagname[@attribute='value']

XPath with OR

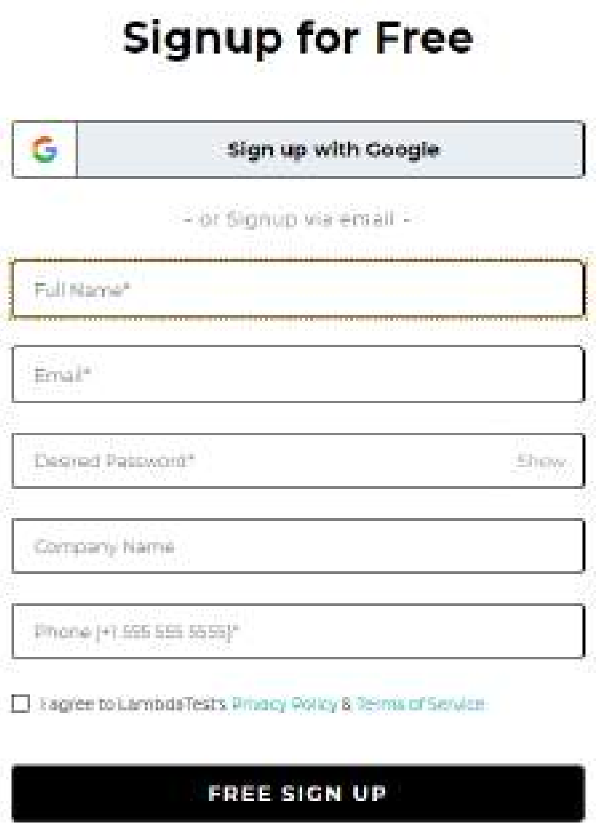


https://accounts.lambdatest.com/register

```
▼<div class="form-group">
    <input type="text" placeholder="Company Name" name=
    "organization_name" value class="form-control " style
    xpath="1"> == $0
  </div>
```

```
//input[@name='organization_name' or @placeholder='Organization/Company Name']
```

XPath with AND


Signup for Free

Sign up with Google

- or Signup via email -

Full Name*

Email*

Desired Password* Show

Company Name

Phone (+1 555 555 5555)*

I agree to LambdaTests Privacy Policy & Terms of Service

FREE SIGN UP

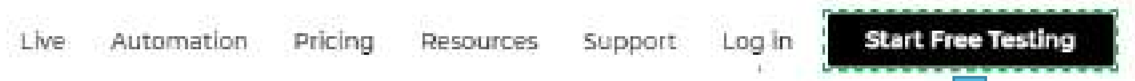
https://accounts.lambdatest.com/register

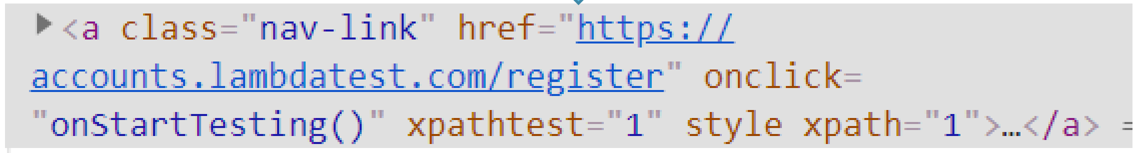
```
▼<div class="form-group">
    <input type="text" placeholder="Full Name*"
    name="name" value required="required" class=
    "form-control " xpath="1"> == $0
 </div>
```

```
//input[@name='name' and @placeholder='Full Name*']
```

XPath with contains()

https://www.lambdatest.com/



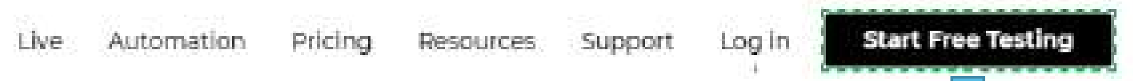


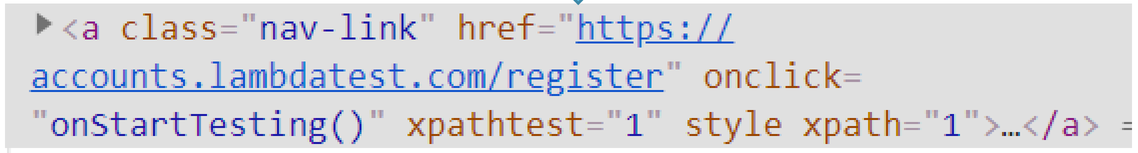
```
▶<a class="nav-link" href="https://
accounts.lambdatest.com/register" onclick=
"onStartTesting()" xpathtest="1" style xpath="1">…</a> =
```

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

//a[contains(@id, ‘value’)]
```

XPath with starts-with()

https://www.lambdatest.com/

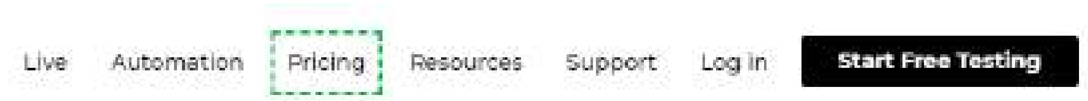




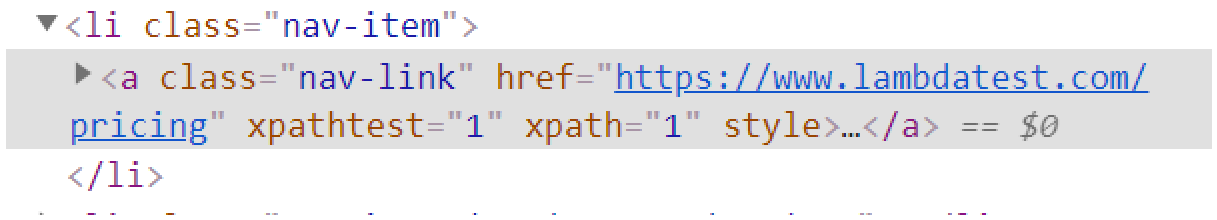
```
▶<a class="nav-link" href="https://
accounts.lambdatest.com/register" onclick=
"onStartTesting()" xpathtest="1" style xpath="1">…</a> =
```

```
//a[starts-with(text(), ‘Start')]

//a[starts-with(@name,‘value')]
```

XPath with Text()



https://www.lambdatest.com/



```
▼<li class="nav-item">
  ▶<a class="nav-link" href="https://www.lambdatest.com/
  pricing" xpathtest="1" xpath="1" style>…</a> == $0
  </li>
```

```
//a[text()='Pricing']
```