**What is XPath in Selenium?**

XPath, also known as XML Path, is one of the most commonly used [locators in Selenium WebDriver](https://www.lambdatest.com/blog/locators-in-selenium-webdriver-with-examples/) that can help you navigate through the HTML structure of a page. It can be used for HTML and XML documents to locate any element in a web page using HTML DOM structure.

The basic format of XPath in Selenium is explained below.

**XPath = //tagname[@Attribute=’Value’]**

Here,

1. **//**: denotes the current node
2. **tagname**: denotes the tagname of the current node
3. **@**: is the Select attribute
4. **Attribute**: denotes the attribute of the node
5. **Value**: denotes the value of the chosen attribute

## Types of XPath in Selenium

The XPath is the language used to select elements in an HTML page. XPath can be used to locate any element on a page based on its tag name, ID, CSS class, and so on. There are two types of XPath in Selenium.

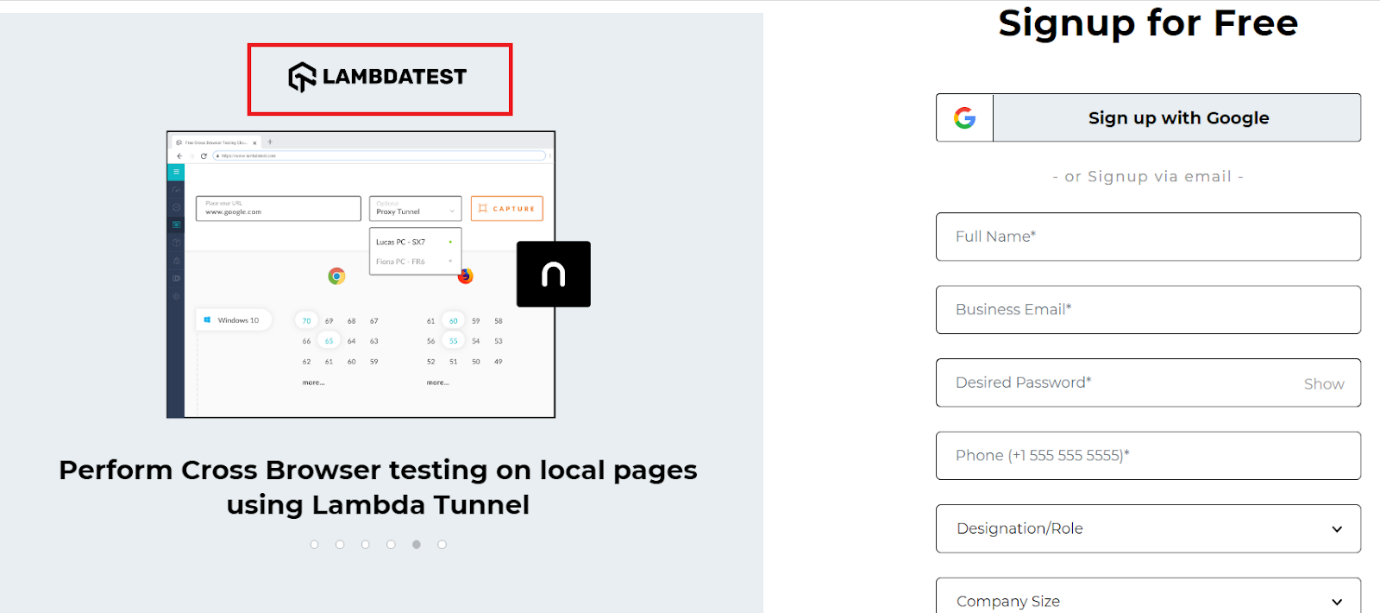
1. Absolute XPath
2. Relative XPath

### **Absolute XPath**

Absolute Xpath is the simplest form of XPath in Selenium. It starts with a single slash ‘/’ and provides the absolute path of an element in the entire DOM.

Let us understand writing an absolute XPath using the [LambdaTest SignUp Page](https://accounts.lambdatest.com/register/" \t "_blank).  
LambdaTest is a cloud-based [cross browser testing](https://www.lambdatest.com/feature) tool that supports Selenium Grid, providing a solution to every obstacle you face while performing automation testing using your local machine. Selenium testing tools like LambdaTest offer an [online Selenium Grid](https://www.lambdatest.com/selenium-grid-online)consisting of 2000+ browsers for you to perform automation testing effortlessly.

We will locate the LambdaTest page header highlighted in the below image using an absolute XPath in Selenium.

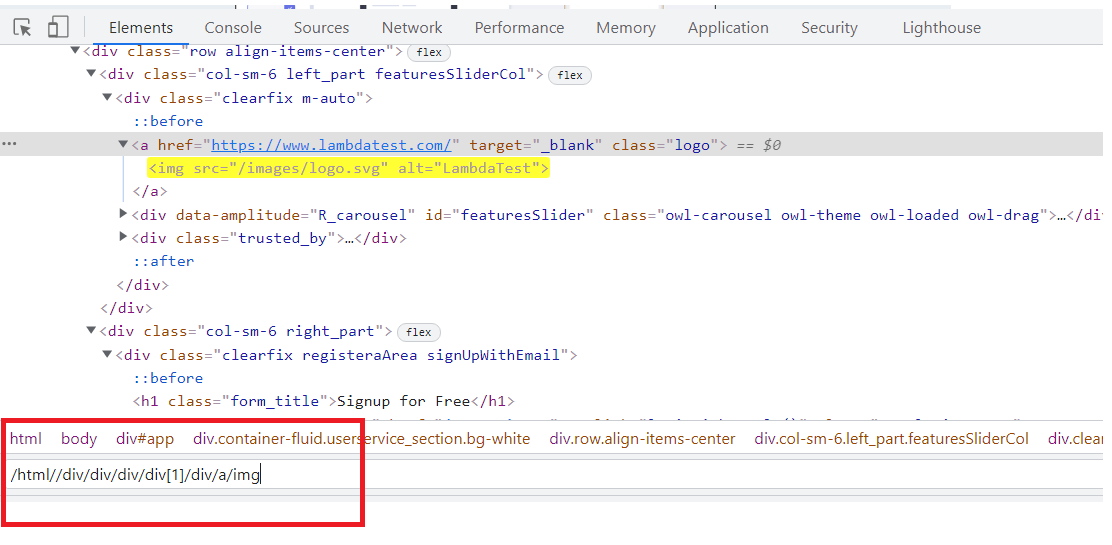


The below absolute XPath will help you locate the header as highlighted.

**/html//div/div/div/div[1]/div/a/img**

In order to locate the element, you can simply do a right-click on the web element and click on Inspect. Then, in the Elements tab, you can start writing the locator.

In Element tab, in chrome browser, use combinations of key i.e. Ctrl + F and start writing the locator.



Way to Take Absolute Xpath

1. Go to the Selector present at the extreme left corner of dev option box.
2. Click on locator icon, then hover mouse on any element you want to choose
3. Automatically, below in Element tab, that element highlighted with yellow color.
4. Right click on that highlighted line.
5. Go to Copy > Choose Copy Xpath or Full Xpath option and paste it in below in locator box.
6. See at the right side that, 1 of 1 is showing or not.

In this case, starting from the html tag, I have traversed one by one to the div, which contains the tag a and hence, the final img tag. Wasn’t it pretty simple?

However, even though it is simple, the biggest disadvantage of using absolute XPath is that they are very vulnerable to any changes in the DOM structure and, as a result, can bring you a lot of automation failures.

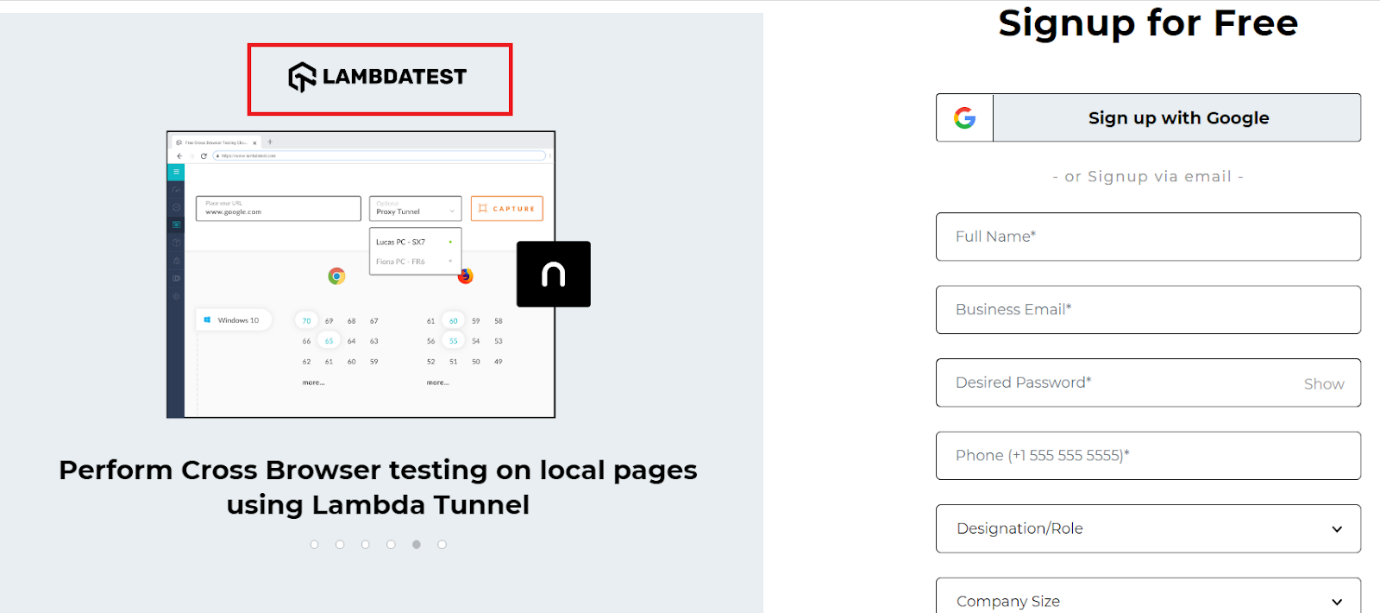
Just imagine if a single div tag from our example is removed, and alas, this locator stops working!!

### **Relative XPath**

In the case of relative XPath in Selenium, the XPath expression starts from the middle of the DOM structure. It is represented by a double slash ‘//’ denoting the current node.

It is always preferred over an absolute XPath as it is not a complete path from the root element.

We will locate the same element as in the case of absolute XPath, i.e., the [LambdaTest Sign Up page](https://accounts.lambdatest.com/register/" \t "_blank) header.



The Relative XPath of the highlighted web element will be:

**//img[@alt=’LambdaTest’]**

Here, I just used the corresponding img tag and its attribute alt to locate the title and wrote the corresponding relative XPath.

### **Difference between Absolute and Relative XPath**

| **Point of Difference** | **Absolute Path** | **Relative Path** |
| --- | --- | --- |
| **Starts with** | Single Forward Slash. Select the element from the root <html> and cover the whole path to the element. It is also known as complete or Full Xpath. | Double Forward Slash. Expression can starts in the middle of the HTML DOM structure. |
| **Speed** | Faster. It identify the element very fast. | Slower compare to absolute. it will take more time in identifying the element as we specify the partial path not (exact path). |
| **Failure Chances** | More. It Changes Frequently, if there are any changes made in the path of the element then XPath gets failed. | Failure chance of well written relative path is very less |
| **Example** | **/html/head/body/form/table/tbody/tr/th**  if any tag will be added before table the path will fail. | **//table/tbody/tr/th**  Doesn't matter if anything added before table. |