XPath Contains, Sibling, Ancestor, AND OR, Parent, Start with, Axes in Selenium Webdriver

If a simple XPath (/xpath-selenium.html) is not able to find a complicated web element for our test script, we need to use the functions from XPath 1.0 library. With the combination of these functions, we can create more specific XPath. Let's discuss a 3 such functions –

1. Contains
2. Sibling
3. Ancestor
4. And OR
5. Parent
6. Starts with
7. XPath Axes

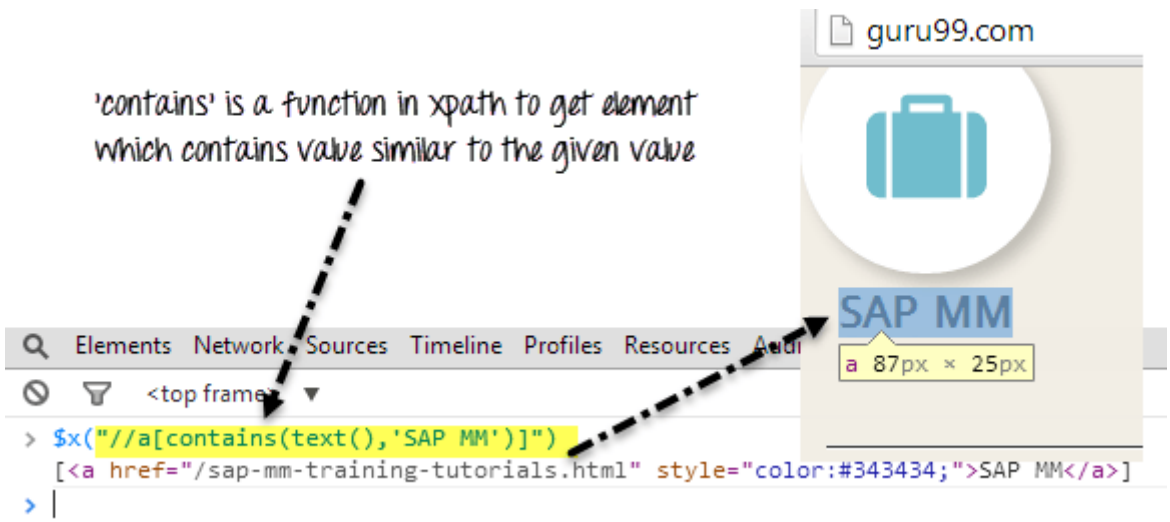
Let's study them in detail -

Contains

By using 'contains' function in XPath, we can extract all the elements which matches a particular text value.

Ex. Here we are searching an anchor .contains text as 'SAP M'.

```
"//h4/a[contains(text(),'SAP M')]"
```



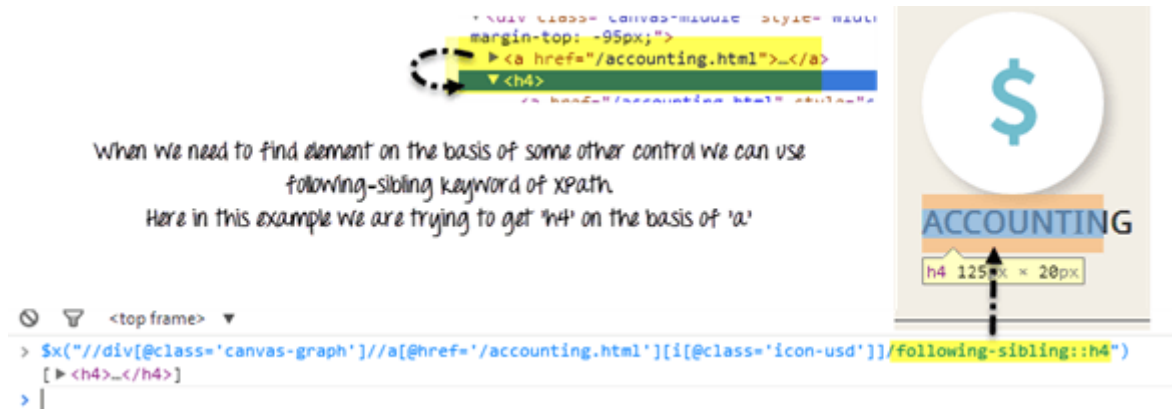
(/images/AdvanceSelenium/071514_0734_UsingContai1.png)

Sibling

Using sibling keyword, we can fetch a web element on the which is related to some other element.

Example: Here on the basis of sibling element of 'a' we are finding 'h4'

```
"//div[@class='canvas- graph']//a[@href='/accounting.html'][i[@class='icon-usd']]/following
-sibling::h4"
```



(/images/AdvanceSelenium/071514_0734_UsingContai2.png)

Ancestor: To find an element on the basis of the parent element we can use ancestor attribute of XPath.

(/images/AdvanceSelenium/071514_0734_UsingContai3.png)

Lets understand these 3 functions using an example –

Test Steps

Note: Since the date of creation of tutorial the Homepage of Guru99 has been updated so use the demo site instead to run tests

1. Go to http://demo.guru99.com/test/guru99home/ (http://demo.guru99.com/test/guru99home/)
2. In the section 'A few of our most popular courses', search all Web Elements which are sibling of a WebElement whose text is 'SELENIUM'

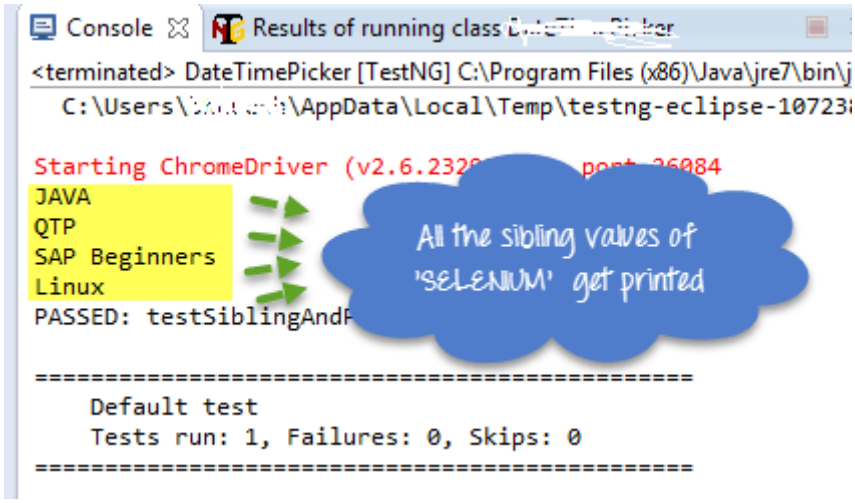
3. We will find element using contains , ancestor and sibling function

(/images/AdvanceSelenium/071514_0734_UsingContai4.png)

USING Contains and Sibling

```
import java.util.List;
import java.util.concurrent.TimeUnit;
import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.WebElement;
import org.openqa.selenium.firefox.FirefoxDriver;
import org.testng.annotations.Test;

public class SiblingAndParentInXpath {

    @Test

    public void testSiblingAndParentInXpath(){

        WebDriver driver;
        String driverPath = "C:\\geckodriver.exe";
        System.setProperty("webdriver.firefox.marionette", driverPath);
        driver = new FirefoxDriver();

        driver.manage().timeouts().implicitlyWait(10, TimeUnit.SECONDS);
        driver.get("http://demo.guru99.com/test/guru99home/");

        //Search element inside 'Popular course' which are sibling of control 'SELENIUM' ,H
ere first we will find a h2 whose text is ''A few of our most popular courses' ,then we mov
e to its parent element which is a 'div' , inside this div we will find a link whose text i
s 'SELENIUM' then at last we will find all of the sibling elements of this link('SELENIUM')

        List <WebElement> dateBox = driver.findElements(By.xpath("//h2[contains(text(),'A f
ew of our most popular courses')]/parent::div//div[//a[text()='SELENIUM']]/following-siblin
g::div[@class='rt-grid-2 rt-omega']"));

        //Print all the which are sibling of the the element named as 'SELENIUM' in 'Popula
r course'
        for (WebElement webElement : dateBox) {
            System.out.println(webElement.getText());
        }

        driver.close();
    }
}
```

Output will be like:



(/images/AdvanceSelenium/071514_0734_UsingContai5.png)

Ancestor function

We can achieve the same functionality with the help of a function 'ancestor' as well.

Now suppose we need to Search All elements in 'Popular course' section with the help of ancestor of the anchor whose text is 'SELENIUM'

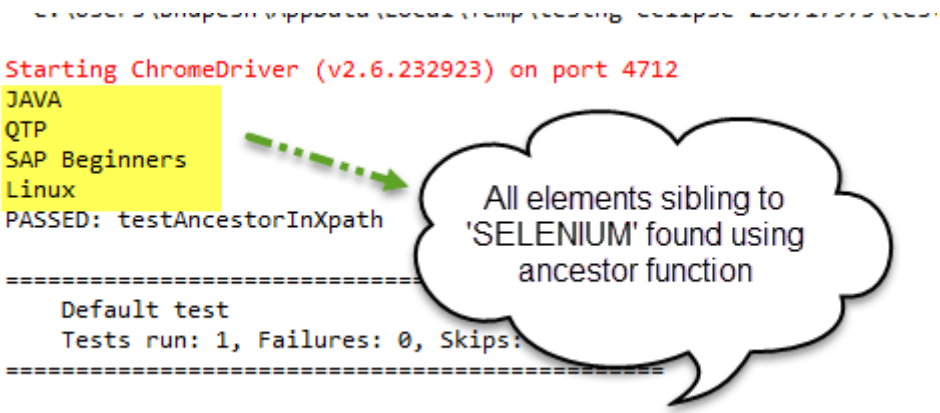
Here our xpath query will be like

```
"//div[.//a[text()='SELENIUM']]/ancestor::div[@class='rt-grid-2 rt-omega']/following-siblin
g::div"
```

Complete Code

```
import java.util.List;
import java.util.concurrent.TimeUnit;
import org.openqa.selenium.By;
import org.openqa.selenium.Keys;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.WebElement;
import org.openqa.selenium.firefox.FirefoxDriver;
import org.testng.annotations.Test;

public class AncestorInXpath{

@Test

    public void testAncestorInXpath(){

        WebDriver driver = new FirefoxDriver();
                driver.manage().timeouts().implicitlyWait(10, TimeUnit.SECONDS);
        driver.get("http://demo.guru99.com/test/guru99home/");

        //Search All elements in 'Popular course' section
                //with the help of ancestor of the anchor whose text is 'SELENIUM'

        List <WebElement> dateBox = driver.findElements(By.xpath("//div[.//a[text()='SELENI
UM']]/ancestor::div[@class='rt-grid-2 rt-omega']/following-sibling::div"));

        //Print all the which are sibling of the element named as 'SELENIUM' in 'Popular co
urse'

        for (WebElement webElement : dateBox) {
            System.out.println(webElement.getText());
        }

        driver.quit();
    }
}
```

Output will look like-



(/images/AdvanceSelenium/071514_0734_UsingContai6.png)

Using AND and OR

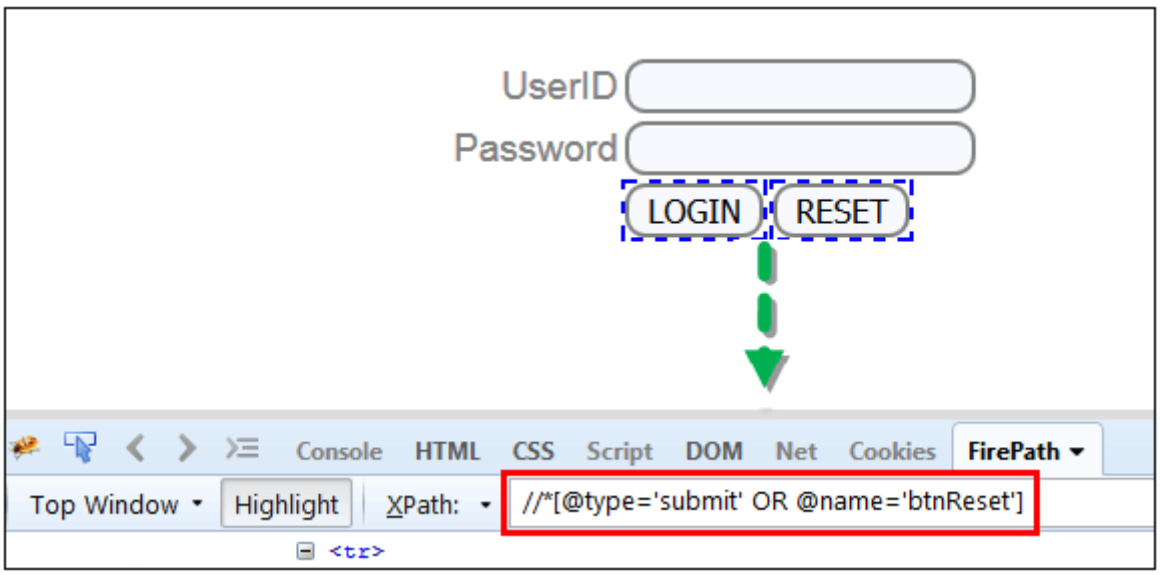
By using AND and OR you can put 2 conditions in our XPath expression.

- In case of AND both 2 conditions should be true then only it finds the element.
- In case of OR any one of the 2 conditions should be true then only it finds the element.

Here our XPath query will be like

```
Xpath=//*[@type='submit' OR @name='btnReset']
```

```
Xpath=//input[@type='submit' and @name='btnLogin']
```



(/images/1/081319_1305_XPathContai7.png)

Test Steps :

1. Go to http://demo.guru99.com/v1/ (http://demo.guru99.com/v1/)
2. In the section, will use the above demo site to search element with different functions of XPath.

You will find an element using AND and OR, parent, starts-with, and XPath axes

AND OR Example

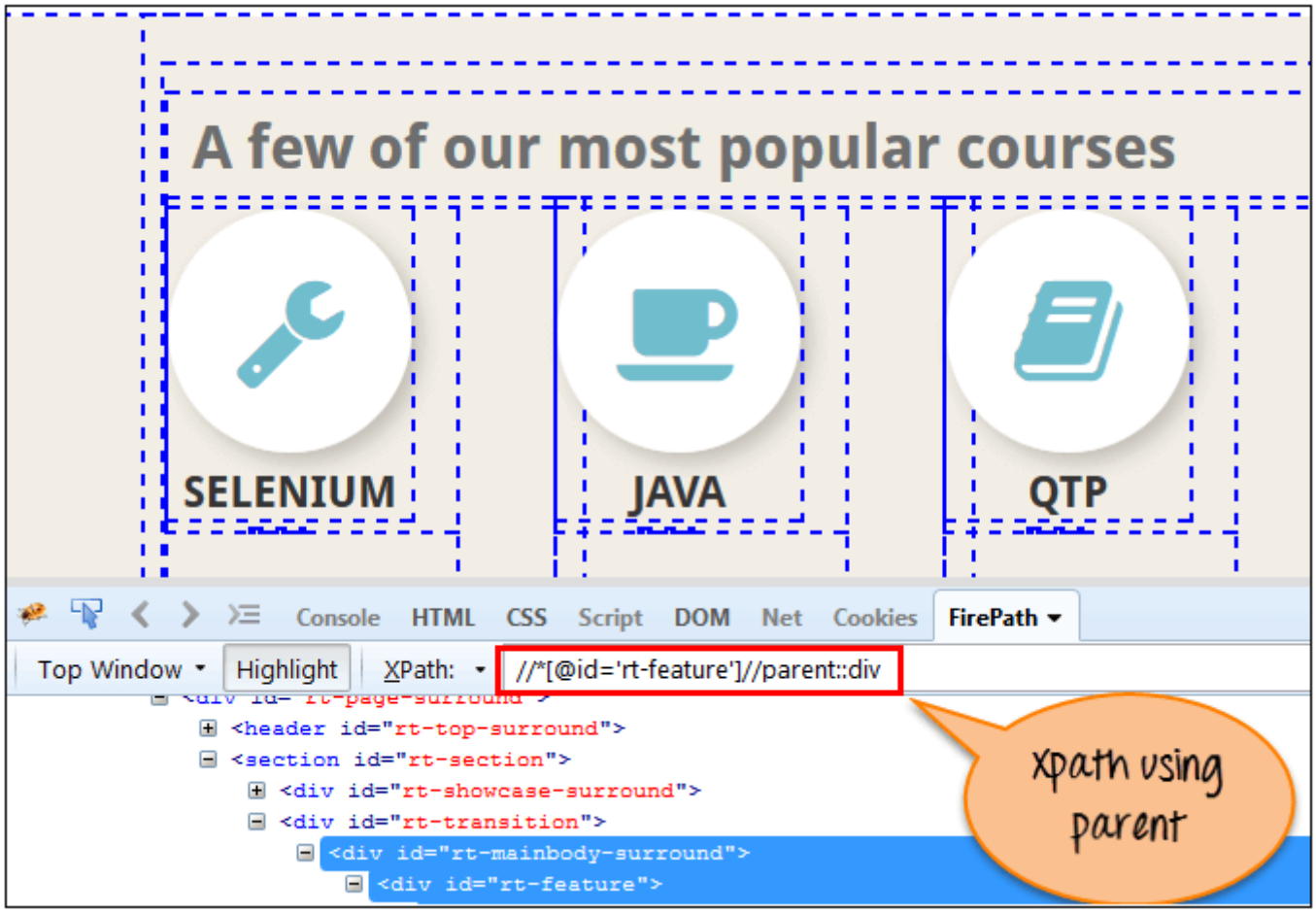
```
import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.WebElement;
import org.openqa.selenium.chrome.ChromeDriver;

public class AND_OR {

        public static void main(String[] args) {
                WebDriver driver;
                WebElement w,x;
                System.setProperty("webdriver.chrome.driver","E://Selenium//Selenium_Jars//
chromedriver.exe");
                 driver= new ChromeDriver();

         // Launch the application
         driver.get("https://www.guru99.com/");

        //Search element using OR in the xpath
         w=driver.findElement(By.xpath("//*[@type='submit' OR @name='btnReset']"));

         //Print the text of the element
                        System.out.println(w.getText());

                //Search element using AND in the xpath
                        x=driver.findElement(By.xpath("//input[@type='submit' and @name='bt
nLogin']"));

                //Print the text of the searched element
                        System.out.println(x.getText());

        //Close the browser
     driver.quit();
        }

}
```

Parent

By Using Parent, you can find the parent node of the current node in the web page.

Here our XPath query will be like

```
Xpath=//*[@id='rt-feature']//parent::div
```



(/images/1/081319_1305_XPathContai8.png)

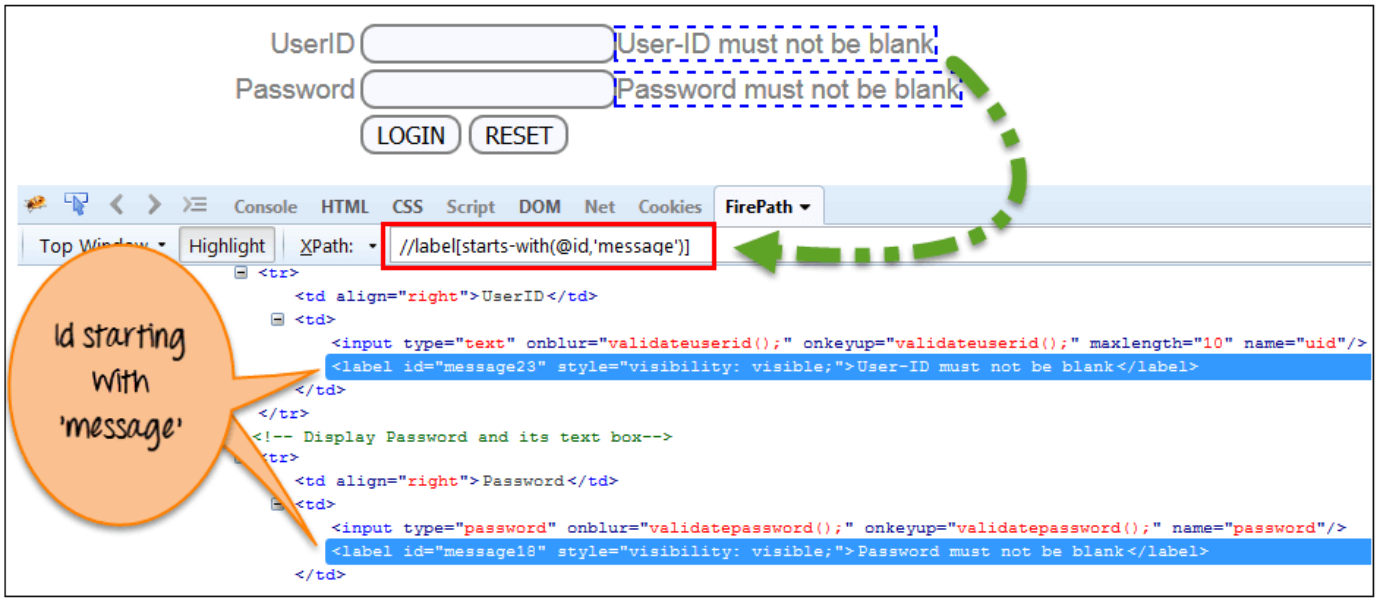
```
import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.WebElement;
import org.openqa.selenium.chrome.ChromeDriver;


public class Parent {

        public static void main(String[] args) {
                WebDriver driver;
                WebElement w;

                System.setProperty("webdriver.chrome.driver","E://Selenium//Selenium_Jars//
chromedriver.exe");
                 driver= new ChromeDriver();

         // Launch the application
         driver.get("https://www.guru99.com/");

         //Search the element by using PARENT
         w=driver.findElement(By.xpath("//*[@id='rt-feature']//parent::div"));

                //Print the text of the searched element
         System.out.println(w.getText());

        //Close the browser
     driver.quit();

        }

}
```

Starts-with

Using Starts-with function, you can find the element whose attribute dynamically changes on refresh or other operations like click, submit, etc.

Here our XPath query will be like

```
Xpath=//label[starts-with(@id,'message')]
```



(/images/1/081319_1305_XPathContai9.png)

```
import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.WebElement;
import org.openqa.selenium.chrome.ChromeDriver;


public class StartsWith {

        public static void main(String[] args) {
                WebDriver driver;
                WebElement w;

                System.setProperty("webdriver.chrome.driver","E://Selenium//Selenium_Jars//
chromedriver.exe");
                 driver= new ChromeDriver();

          // Launch the application
          driver.get("https://www.guru99.com/");

          //Search the element by using starts-with
          w=driver.findElement(By.xpath("//label[starts-with(@id,'message')]"));

          //Print the text of the searched element
        System.out.println(w.getText());

        //Close the browser
                driver.quit();
        }

}
```

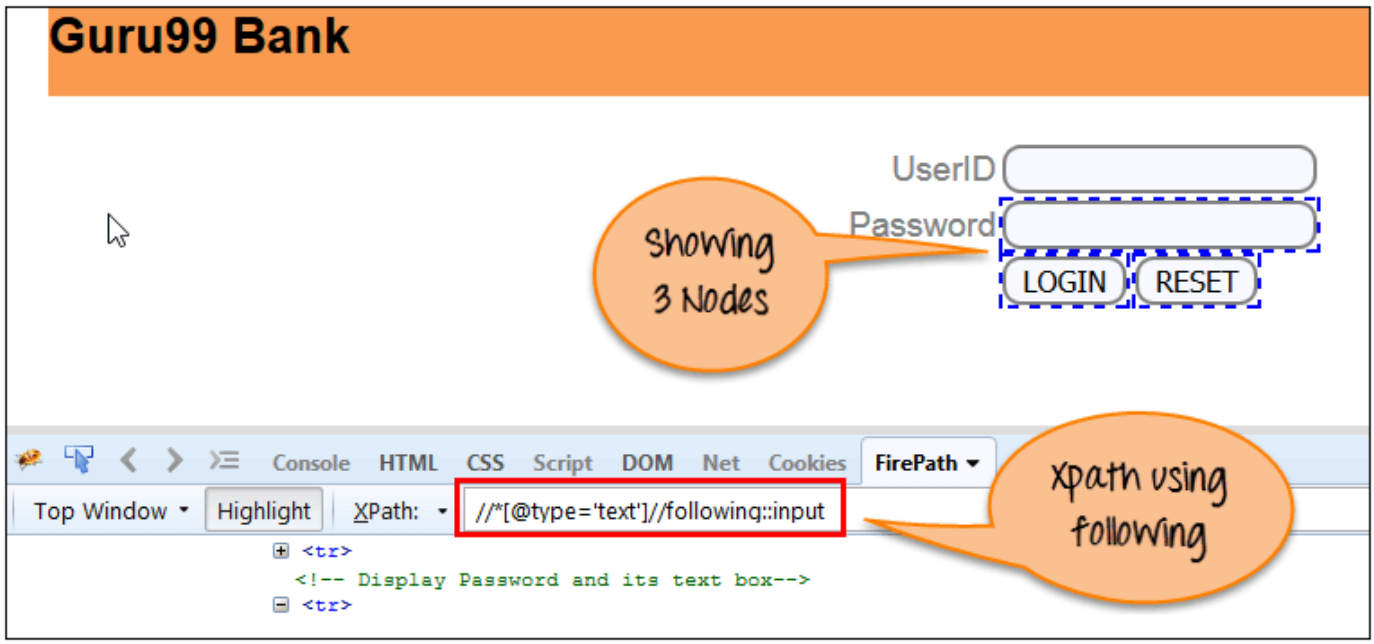
Xpath axes

By using XPath axes, you can find the dynamic and very complex elements on a web page. XPath axes contain several methods to find an element. Here, will discuss a few methods.

following: This function will return the immediate element of the particular component.

Here our XPath query will be like

```
Xpath=//*[@type='text']//following::input
```



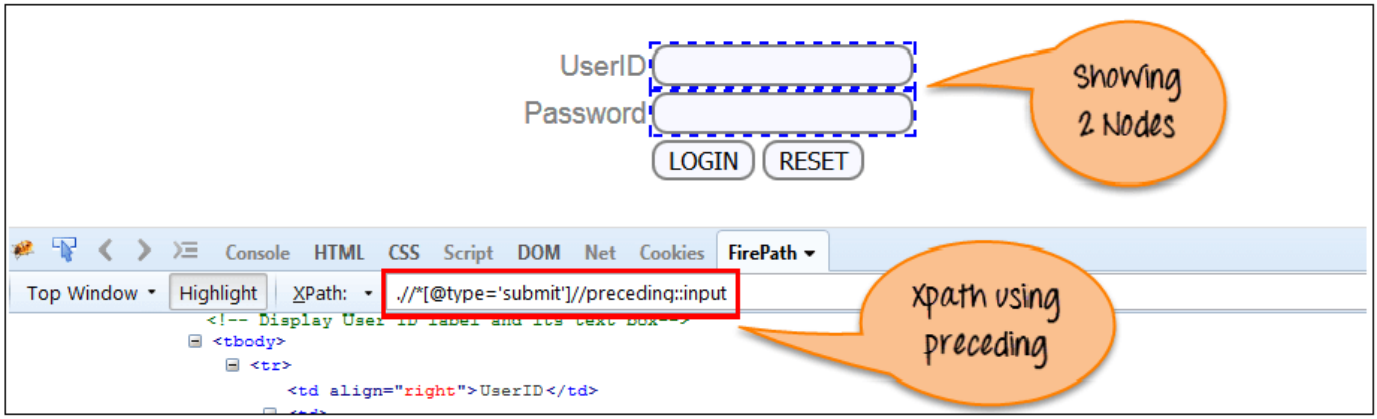
(/images/1/081319_1305_XPathContai10.png)

```
import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.WebElement;
import org.openqa.selenium.chrome.ChromeDriver;


public class Following {

        public static void main(String[] args) {
                WebDriver driver;
                WebElement w;

                System.setProperty("webdriver.chrome.driver","E://Selenium//Selenium_Jars//
chromedriver.exe");
                 driver= new ChromeDriver();

          // Launch the application
          driver.get("https://www.guru99.com/");

          //Search the element by using Following method
          w=driver.findElement(By.xpath("//*[@type='text']//following::input"));

                //Print the text of the searched element
          System.out.println(w.getText());

         //Close the browser
     driver.quit();
        }

}
```

Preceding: This function will return the preceding element of the particular element.

Here our XPath query will be like

```
Xpath= //*[@type='submit']//preceding::input
```



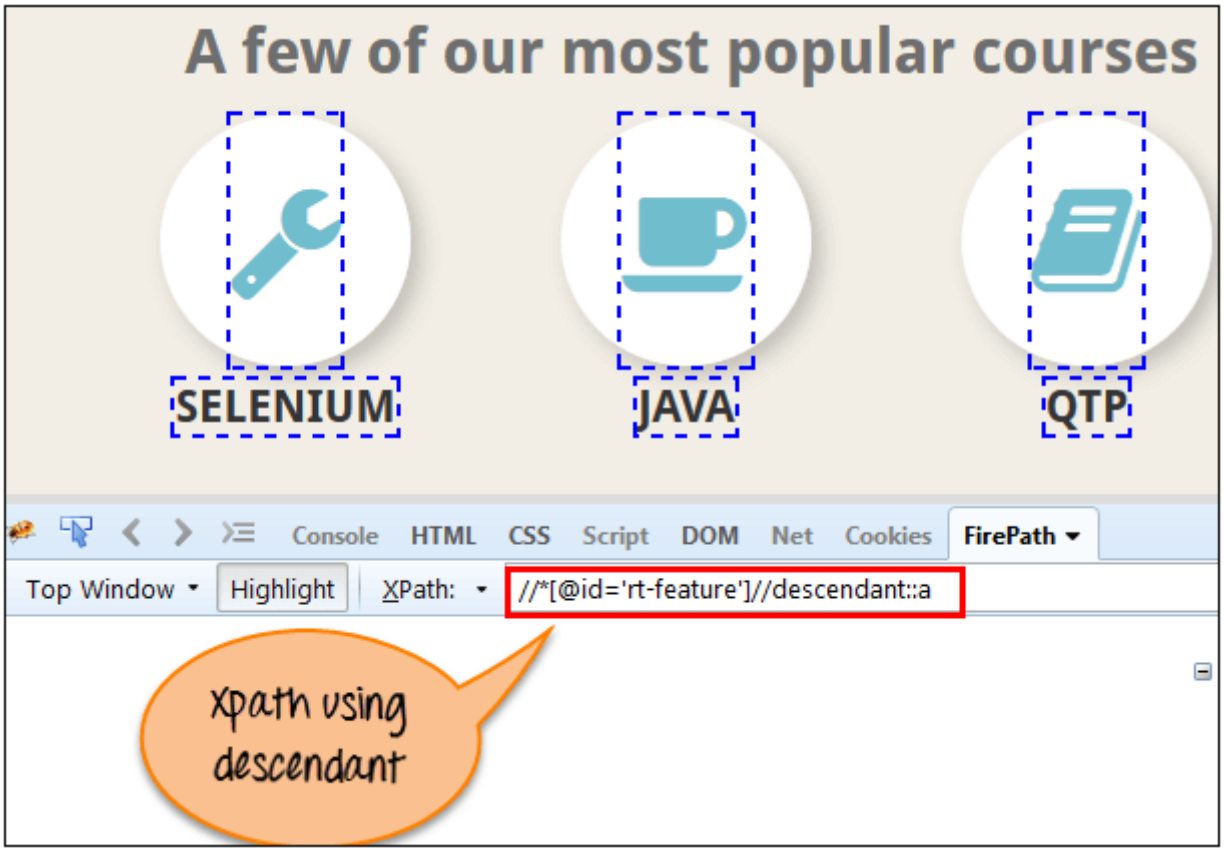
(/images/1/081319_1305_XPathContai11.png)

```
import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.WebElement;
import org.openqa.selenium.chrome.ChromeDriver;


public class Preceding {

        public static void main(String[] args) {

                WebDriver driver;
                WebElement w;

                System.setProperty("webdriver.chrome.driver","E://Selenium//Selenium_Jars//
chromedriver.exe");
                 driver= new ChromeDriver();

         // Launch the application
         driver.get("https://www.guru99.com/");

         //Search the element by using preceding method
         w=driver.findElement(By.xpath("//*[@type='submit']//preceding::input"));

                //Print the searched element
         System.out.println(w.getText());

        //Close the browser
     driver.quit();

        }

}
```

d) **Descendant:** This function will return the descendant element of the particular element.

Here our XPath query will be like

```
Xpath= //*[@id='rt-feature']//descendant::a
```



(/images/1/081319_1305_XPathContai12.png)

```
import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.WebElement;
import org.openqa.selenium.chrome.ChromeDriver;


public class Descendant {

        public static void main(String[] args) {
                WebDriver driver;
                WebElement w;
                System.setProperty("webdriver.chrome.driver","E://Selenium//Selenium_Jars//
chromedriver.exe");
                 driver= new ChromeDriver();

         // Launch the application
         driver.get("https://www.guru99.com/");

         //Search the element by using descendant method
         w=driver.findElement(By.xpath("//*[@id='rt-feature']//descendant::a"));

                //Print the searched element
         System.out.println(w.getText());

        //Close the browser
     driver.quit();

        }

}
```

Summary

- There are some situations when regular XPath cannot be used to find an element. In such situation, we need different functions from the xpath query.
- There some important XPath functions like contains, parent, ancestors, following-sibling, etc.
- With the help of these functions, you can create complex XPath expressions.

Prev Report a Bug Next

(/maximize-resize-minimize-browser-selenium.html)

(/maximize-resize-minimize-browser-selenium.html)

Maximize Browser in Selenium

(/maximize-resize-minimize-browser-selenium.html)

(/handling-dynamic-selenium-webdriver.html)

(/handling-dynamic-selenium-webdriver.html)

Handling Dynamic Web Tables Using Selenium WebDriver

(/handling-dynamic-selenium-webdriver.html)

(/selenium-alternatives.html)

(/selenium-alternatives.html)

Top 15 Selenium Alternatives in 2019

(/selenium-alternatives.html)

SELENIUM

(/chrome-options-desiredcapabilities.html)

(/chrome-options-desiredcapabilities.html)

Chrome Options & Desiredcapabilities: AdBlocker, Incognito, Headless

(/chrome-options-desiredcapabilities.html)

SELENIUM

(/gecko-marionette-driver-selenium.html)

(/gecko-marionette-driver-selenium.html)

Gecko (Marionette) Driver Selenium: Download, Install, Use with Firefox

(/gecko-marionette-driver-selenium.html)

SELENIUM

(/scroll-up-down-selenium-webdriver.html)

(/scroll-up-down-selenium-webdriver.html)

How to Scroll Down or UP a Page in Selenium Webdriver

(/scroll-up-down-selenium-webdriver.html)

Selenium Tutorials

(https://www.facebook.com/guru99com/)
(https://twitter.com/guru99com)
(https://www.youtube.com/channel/UC19i1XD6k88KqHlET8atqFQ)
(https://forms.aweber.com/form/46/724807646.htm)

About

About Us (/about-us.html)
Advertise with Us (/advertise-us.html)
Write For Us (/become-an-instructor.html)
Contact Us (/contact-us.html)

Career Suggestion

SAP Career Suggestion Tool (/best-sap-module.html)
Software Testing as a Career (/software-testing-career-complete-guide.html)

Interesting

Books to Read! (/books.html)
Blog (/blog/)
Quiz (/tests.html)
eBook (/ebook-pdf.html)

Execute online

Execute Java Online (/try-java-editor.html)
Execute Javascript (/execute-javascript-online.html)
Execute HTML (/execute-html-online.html)
Execute Python (/execute-python-online.html)

Privacy Policy (/privacy-policy.html) | Affiliate Disclaimer (/affiliate-earning-disclaimer.html)