



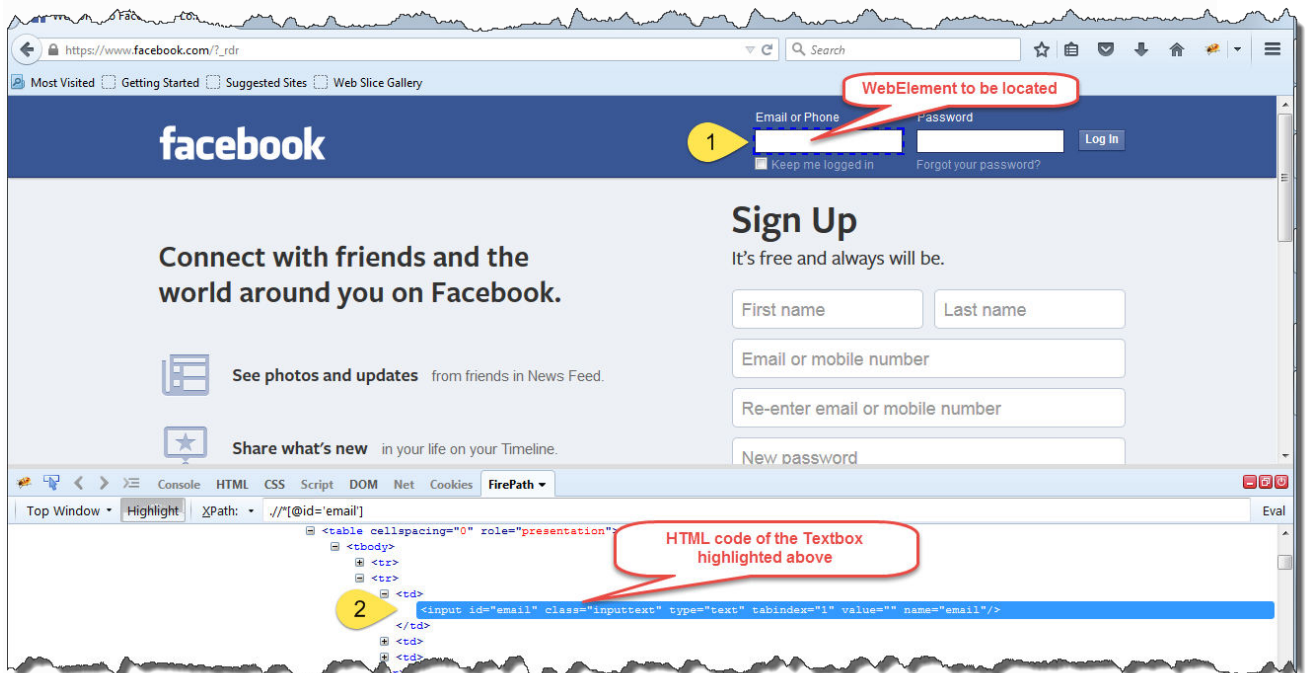
Locating Web Elements in Selenium

There are various strategies to locate elements in a page. You can use the most appropriate one for your case. Selenium provides the following methods to locate elements in a page:

- *find_element_by_id*
- *find_element_by_name*
- *find_element_by_class_name*
- *find_element_by_tag_name*
- *find_element_by_css_selector*
- *find_element_by_XPath*
- *find_element_by_link_text*
- *find_element_by_partial_link_text*

Lets study each technique in detail.

Consider the below image showing the HTML information of text box for Email or Phone we normally uses on Facebook for Log-in.



Now we will use each of the above techniques to locate this web element.

Important thing which we have to keep in mind is , whatever the technique we are using , it must be able to locate the element uniquely.

A) By ID:-



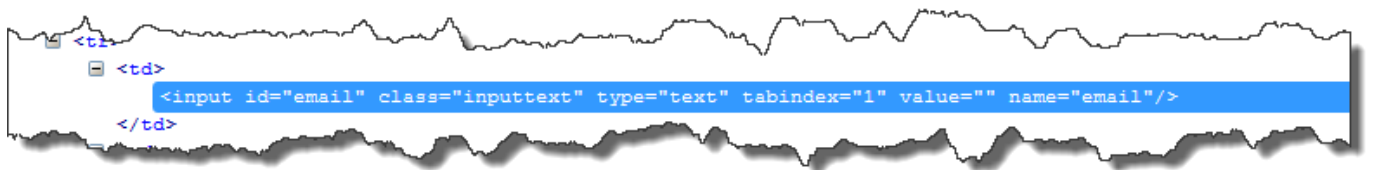
```
dr.find_element_by_id('email').send_keys('thetestingworld@gmail.com')
```

B) By Name :-

```
dr.find_element_by_name('email').send_keys('thetestingworld@gmail.com')
```

C) By Class Name :-

```
dr.find_element_by_class_name('inputtext').send_keys('thetestingworld@gmail.com')
```

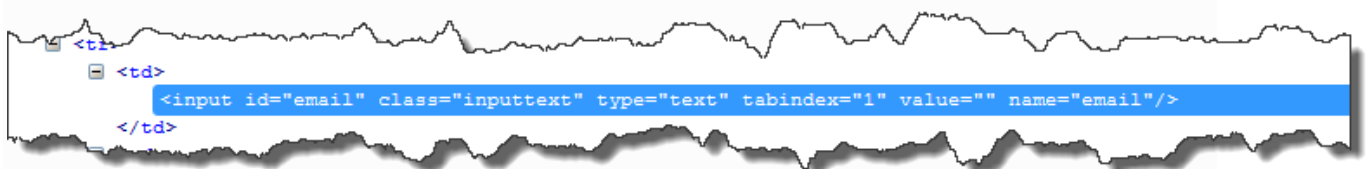
D) By Tag Name :-

```
dr.find_element_by_tag_name('input').send_keys('thetestingworld@gmail.com')
```

This command may not be able to locate this web element uniquely as there could be so many other web elements on the page having the same tag name.

E) By CSS Selector :-

We can use this technique in 4 different ways, which are as follows :-



a. Tagname with ID:-*Css=tagname#ID**Css=input#email*

```
dr.find_element_by_css_selector('input#email').send_keys('thetestingworld@gmail.com')
```

b. Tagname with CLASS:-*Css=tagname.classvalue**Css=input.inputtext*

```
dr.find_element_by_css_selector('input.inputtext').send_keys('thetestingworld@gmail.com')
```

c. Tagname with Attribute:-*Css=tagname[attribute=value]**Css=input[tabindex='1']*

```
dr.find_element_by_css_selector("input[tabindex='1']").send_keys('thetestingworld@gmail.com')
```

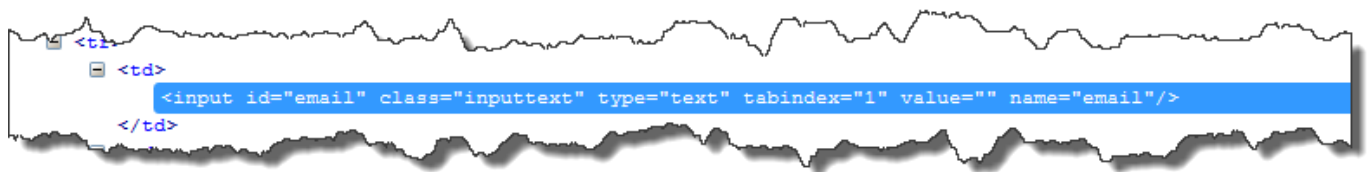
d. Tagname with Class and Attribute:-*Css=tagname.classvalue[attribute=value]**Css= input.inputtext[tabindex='1']*

```
dr.find_element_by_css_selector("input.inputtext[tabindex='1']").send_keys('thetestingworld@gmail.com')
```

F) By XPath:-

XPath provides the hierarchical path of the web element from top to the bottom.

There are so many ways to use XPath :-

**A) Simple XPath :-***XPath = absolute or relative XPath**XPath = //label/input (RP) or**XPath = /html/body/div/div[2]/div/div/div/form/div/fieldset/label/input (AP)*

```
dr.find_element_by_xpath("./*/table/tbody/tr[2]/td[1]").send_keys('thetestingworld@gmail.com')
```

B) *XPath with Attribute(s) :-*

XPath=AP or RP[@attribute1='value' and/or @attribute2='value' and/or]

XPath=//input[@id='email']

```
dr.find_element_by_xpath("//input[@id='email']").send_keys('thetestingworld@gmail.com')
```

C) *XPath with Position (indexing) :-*

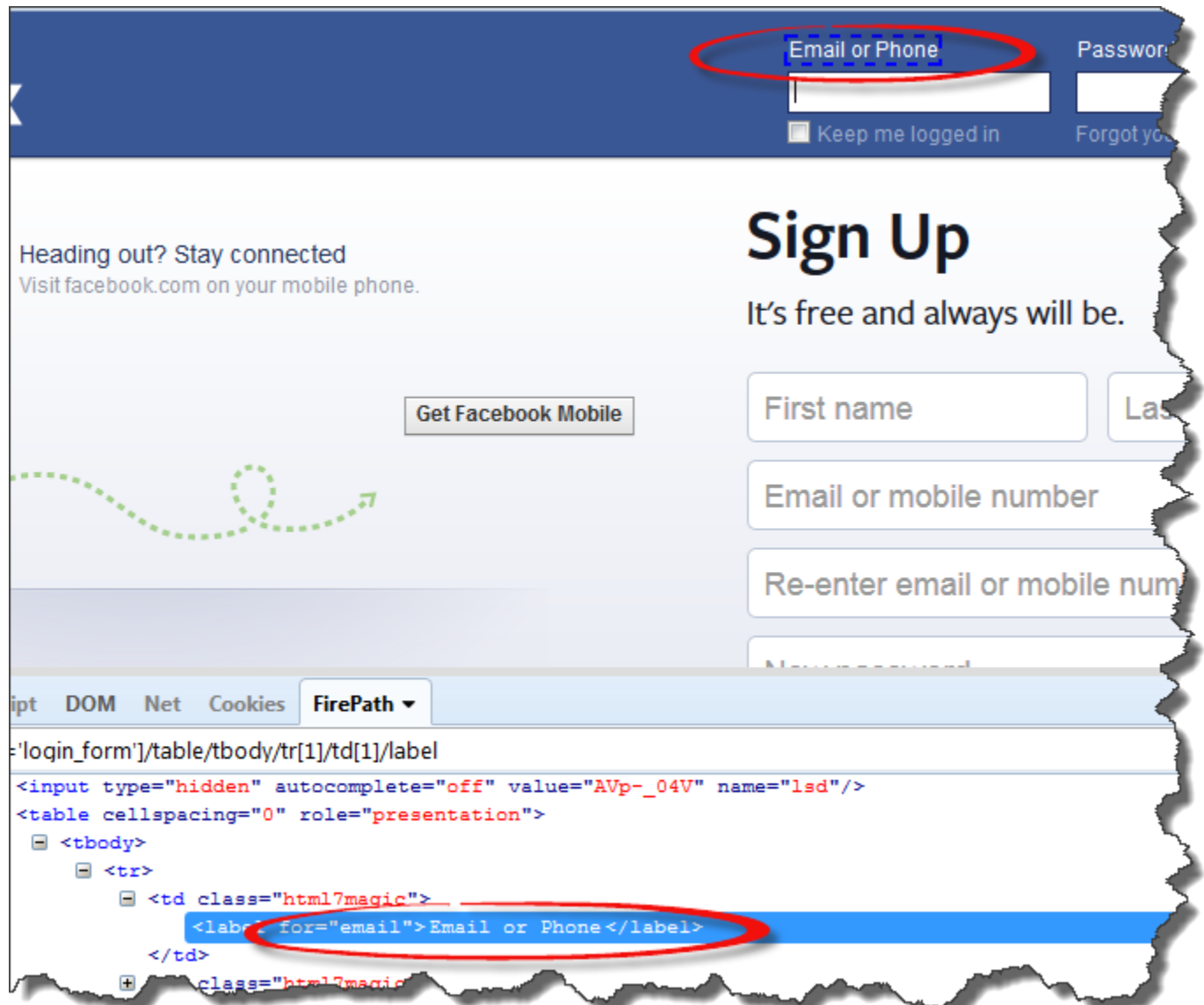
XPath=(AP or RP[@attribute='value'])[N]

Where N is the index

XPath=(./input)[2]

```
dr.find_element_by_xpath("(./input)[2]").send_keys('thetestingworld@gmail.com')
```

Consider the below image of locating a hyperlink 'Email or Phone' on login page of Facebook.



We can click on hyperlinks using any of the below mentioned technique.

G) By Link Text:-

```
dr.find_element_by_link_text("Email or Phone").  
send_keys('thetestingworld@gmail.com')
```

H) By Partial Link Text :-

```
dr.find_element_by_partial_link_text("Email or").  
send_keys('thetestingworld@gmail.com')
```