

Locators in Selenium WebDriver

Locating elements in WebDriver is done by using the method **“findElement(By.locator())”**.

We use above mentioned method to locate elements but the locator() part will be replaced with the locator names. Mentioned few examples below for reference.

```
findElement(By.id("IdName"))  
findElement(By.className("ClassName"))
```

Let's see each of the Locators in Selenium in detail:

ID Locator:

ID's are supposed to be unique on a page and it makes IDs the most reliable locator. ID locators are the fastest and safest locators out of all locators.

id = id of the element

```
findElement(By.id("IdName"))
```

Name Locator:

We sometimes use **Name** locator to identify the elements on our webpage. Locating elements using Name is same as locating elements using ID locator.

These are not unique on a page. If there are multiple elements with the same Name locator then the first element on the page is selected. Test may fail, if another element with the same Name locator is present on the web page or added by the developers in the later stages.

Name = Name of the element

```
findElement(By.name("Name"))
```

Class Name Locator:

Class Name locator gives the element which matches the values specified in the attribute name "class".

```
findElement(By.className("Element Class"))
```

Tag Name Locator:

Tag Name locator is used to find the elements matching the specified Tag Name. It is very helpful when we want to extract the content within a Tag.

```
findElement(By.tagName("HTML Tag Name"))
```

Link Text Locator:

If there are multiple elements with the same link text then the first one will be selected. This Link Text locator works only on links (hyperlinks) so it is called as Link Text locator.

```
findElement(By.linkText("LinkText"))
```

Partial Link Text:

In some situations, we may need to find links by a portion of the text in a Link Text element. it contains. In such situations, we use Partial Link Text to locate elements.

```
findElement(By.partialLinkText("partialLinkText"))
```

CSS Selector Locator:

There is a debate on the performance between CSS Locator and XPath locator. Most of the automation testers believe that using CSS selector makes the execution of script faster compared to XPath locator. This locator is always the best way to locate elements on the page.

Following are the some of the mainly used formats of CSS Selectors.

1- Tag and ID

```
findElement(By.cssSelector(tag#id))
```

2- Tag and Class

```
findElement(By.cssSelector(tag.class))
```

3- Tag and Attribute

```
findElement(By.cssSelector(tag[attribute=value]))
```

4- Tag, Class and Attribute

```
findElement(By.cssSelector(tag.class[attribute=value]))
```

XPath Locator:

XPath is designed to allow the navigation of XML documents, with the purpose of selecting individual elements, attributes, or some other part of an XML document for specific processing. XPath produces reliable locators but in performance wise it is slower (especially in IE older versions) compared to CSS Selector.

```
findElement(By.xpath("XPath"))
```