

DAY 3 LOCATOR NOTES

To be able to do any action on any web element, there are certain steps we must follow:

1. We must locate the web element we want to work on.
2. To locate we must use a set of methods coming from Selenium library.
 - findElement()
 - locator methods
3. We decide what action we want to apply, find the method, and use it on the already located web element.
 - click();
 - sendKeys("");

What is a locator?

- Locators help us locate web elements using Selenium WebDriver.
- Locators are basically 8 methods coming from Selenium library.
- There are total of 8 locators.
- We will learn first 6 today.

.findElement(LOCATOR) method:

1. This method finds and returns SINGLE web element.
2. It accepts a locator argument and finds the requested web element using locator.
3. Return type: WebElement type (comes from Selenium library)
4. It will throw "NoSuchElementException"
 - if you provide a wrong locator
 - if your page does not load as fast as it should

id

- it uses "id" attribute's value to locate web elements.
- it will go through HTML code and find matching "id" attribute value, and return the web element
- id HAS TO BE UNIQUE.
- id attribute's value will always be unique.

```
<div name="hu58" id="asdf57"> </div>
```

```
syntax: driver.findElement(By.id("asdf57"));
```

name

- it uses "name" attribute's value to locate web elements.
- it will go through HTML code and find matching "name" attribute value, and return the first matching web element
- name does not have to be unique. So, use whenever possible.

```
<div name="hu58"> </div>
```

```
syntax: driver.findElement(By.name("hu58"));
```

className

- it uses "class" attribute's value to locate web elements.
- it will go through HTML code and find matching "class" attribute value, and return the first matching web element.
- class attribute's value does not have to be unique.

```
<div name="hu58" id="asdf57" class="cde45"> </div>
```

```
syntax: driver.findElement(By.className("cde45"));
```

tagName

- it will find and return the first matching tag provided.
- this locator does not have many use cases unless either the page is very simple page, or we are trying to get a group of web elements using findElements method

```
<div name="hu58" id="asdf57" class="cde45"> </div>
```

```
syntax: driver.findElement(By.tagName("div"));
```


linkText

- linkText locator will look through the TEXTS OF all the links in the page and return the matching link.
- It accepts a string as the text of the link we are trying to locate
- This locator will ONLY ONLY ONLY work on links. <a> tags only.
- This method looks for EXACT text match.
- It is similar to .equals method coming from JAVA.
- It will search through HTML code and return first matching result.

ex: GOOGLE SEARCH

syntax: driver.findElement(By.linkText("GOOGLE SEARCH"));

partialLinkText

- It will look through the TEXTS OF all the links in the page and return the PARTIAL matching link.
- It accepts a string as the text of the link we are trying to locate
- This locator will ONLY ONLY ONLY work on links. <a> tags only.
- This method looks for PARTIAL text match.
- It is similar to .CONTAINS method coming from JAVA.
- It will search through HTML code and return first matching result.

ex: GOOGLE SEARCH

syntax1: driver.findElement(By.partialLinkText("GOOGLE"));

syntax2: driver.findElement(By.partialLinkText("SEARCH"));

syntax3: driver.findElement(By.partialLinkText("GOOGLE SEARCH"));

All 3 syntaxes returns same web element.

.sendKeys("string to be sent");

- This method is used to send text using Selenium WebDriver
- It accepts a string argument.
- It takes the String and passes it into the web element as a String.