

DAY 4 - CLASSNOTES :

- TODAY SCHEDULE

#1- XPATH - cssSelector

#2- Tasks 1_2_3

#3- StaleElementReferenceException

#4- findElement vs findElements

#5- checkboxes and radiobuttons

- cssSelector:

- **cssSelector** is one of 8 locators of Selenium.
- It allows users/us to create custom locators using attributes and values.
- It does not limit us using only class, id, name attributes.
- We can use ANY attribute and their values.
- It has its own syntax that we must follow to create **cssSelector**.
- Using **cssSelector** we can go from parent to child element.
- To go from parent to child we use **>** sign.
- Using **cssSelector** we cannot go from child to parent.

parentElement > childElement

div[class='large-6 small-12 columns'] > label

- SYNTAX #1:

- We are not limited with any attribute or value.

tagName[attribute='value'] --> more specific way of

writing css

[attribute='value'] --> we can even not provide tagName, but it will be less specific

- SYNTAX #2:

- We are limited with id, and class attributes' values.

-What does . stand for in cssSelector?

- . stands for class in cssSelector

-What does # stand for in cssSelector?

- # stands for id in cssSelector

```
<div id="55k" name="kgl33" for="randomV12">  
  <a href="https://tesla.com"> CYBERTRUCK  
OFFICIAL </a>  
</div>
```

ex#1_syntax#1: write css selector that is locating the a tag

```
a[href='https://tesla.com']
```

ex#2_syntax#1: write css selector that is locating the "div" tag "for" attribute

```
tagName[attribute='value']  
div[for='randomV12']  
div[name='kgl33']  
div[id='55k']
```

- all of the above locators are locating the same web element, <div>.

ex#3_syntax#2:write css selector that is locating the "div" tag "id" attribute

**tagName#idValue
div#55k**

- XPATH LOCATORS

- xpath is one of 8 locators of Selenium
- xpath allows us to create custom locators using provided attributes and their values
- we can also use the text of the provided web element to create locators

- XPATH has 2 different types

Interview question: What is the difference between absolute xpath and relative xpath?

#1- ABSOLUTE XPATH:

- Absolute xpath starts with single slash "/"
- It starts looking in html from the root/parent element : html element
- It starts from html tag, and it goes down 1 by 1 until we reach to the web element we are looking for
- This is not good way of locating a web element.
- It will break with any minimal change in the html code.

/html/body/table/tbody/tr[2]/td/div/div/form/div[4]/button

#2- RELATIVE XPATH

- Relative xpath starts with double slash "//"
- "//" means you can start from anywhere in the HTML code

- Since we are allowed to start from anywhere in the HTML code, relative xpath is very dependable
- We will use relative xpath, not absolute xpath
- The only time your relative xpath is breaking (not working) is when/if the developer is specifically changing the attribute value we used

--> MAIN SYNTAX: `//tagName[@attribute='value']`

- How do we go from parent to child using xpath?
 - Using "/" single slash we can go from parent to direct child.
 - Using "/" double slash we can go from parent to any child (grand child, grand-grand child)

COMMONLY USED XPATH SYNTAXES:

- #1- `//tagName[@attribute='value']`
- #2- `//tagName[contains(@attribute, 'value')]`
- #3- `//tagName[.='text']` same as `//tagName[text()='text']`
- #4- `//*[@attribute='value']`

--> How to move lines up and down without having to copy paste:

- Keep your cursor in the line you want to move
- MAC: command + shift + up or down arrow
- Windows : control + shift + up or down arrow

findElement vs findElements()

- findElement

- returns single web element
- return type: WebElement (coming from Selenium)
- it will throw NoSuchElementException if it cannot find a web element with given locator.

- findElements:

- returns List of WebElements
- return type : List<WebElement>
- it will NOT throw exception, it will return empty list

- driver.manage().timeouts().implicitlyWait(10, TimeUnit.SECONDS);

- Basically this line will allow our driver to wait UP TO given time if driver cannot immediately locate a web element.

- We write it one time, and it is applied to the whole driver session.

- Driver does not have to wait until 10 second is ended. If it finds the web element before maybe in 2 second, 5 seconds it will continue with execution.

- checkboxes & radiobuttons

- How do you handle checkboxes and radio buttons using Selenium?

#1- We can click

**#2- We can check if they are selected or not. --> -
isSelected();**

**#3- We can check if they are enabled or not. --> -
isEnabled();**

**- isSelected(); --> if web element is selected, it will
return true, if not false.**

**- isEnabled(); --> if web element is enabled, it will
return true, if not false.**

Please watch the short videos.

23-28 -> cssSelector and xpath

16-17 -> findElements

19,20 -> checkboxes and radio buttons