12-FEB-2020

===========

Ex: Find Absolute xpath for "Create Account" link in Gmail-Login page

Sol: /html/body/div/div/div[2]/div/div[2]/div/div/div[2]/div/div[2]/div/div/div[2]/div/div/span/span

-- Whenever there is any change in the application

-- If they add any component then Absolute xpath hierarchy will change

-- In general Absolute xpath is not preferred

Where as if any changes are made in a page then webelement absolute xpath also will change, due to that reason WD is unable to identify web element.

--> We will be using Relative xpath in real time

===================================================================================================================================================================

b. Relative xpath [Partial xpath]

==============

Also called as partial xpath or customized xpath.

In general we prefer relative xpath to address webelement.

\*\*IQ. What is Relative xpath?

Relative xpath is a combination of tagname and any one of the attribute value of webelement

Syntax:

------

//tagname[@attributename='value']

Ex:

Find Relative xpath for "Email" edit box in Gmail home page

//input[@id='identifiedId']

(or)

//input[@name='identifier']

(or)

//input[@type='email']

--> Do not prefer classname in this as there are 2 elements found

===================================================================================================================================================================

\*\*\*Q. Difference between Absolute xpath & relative xpath

===================================================

Absolute xpath Relative xpath

============== ==============

1. Starts with ---> / 1. Starts with ----> "//"

2. We don't use any attribute 2. We can use one or more attributes

3. Also called as Full xpath 3. Also called as Partial xpath

===================================================================================================================================================================

Ex: Write script to perform login operation in Orange HRM project using Relative xpath as locator

Script: //To initialize browser

System.setProperty("webdriver.chrome.driver", "./Drivers//chromedriver.exe");

WebDriver driver= new ChromeDriver();

driver.get("https://login.salesforce.com");

driver.manage().window().maximize();

//to perform login operation

driver.findElement(By.xpath("//input[@id='txtUsername']")).sendKeys("Admin");

driver.findElement(By.xpath("//input[@id='txtPassword']")).sendKeys("admin123");

driver.findElement(By.xpath("//input[@id='btnLogin']")).click();

==================================================================================================================================================================

Ex: # Find Relative xpath for "Create Account" web element in Gmail Login Page

//span[@class='N/Wrkb snByac']

==================================================================================================================================================================

DIFFERENT FUNCTIONS/METHODS IN RELATIVE XPATH

=============================================

I. Using "\*" in Relative xpath:

===========================

Syntax: //\*[@attribute='value']

Here "\*" represents any tagname

Ex: //\*[@name='identifier']

[Relative xpath for "Email" editbox]

==================================================================================================================================================================

ii. xpath with multiple Attributes:

==============================

Using Relative xpath we can give more than one attribute value to address web element

Syntax: //tagname[@attribute='value'][@attribute2='value'].....

Ex: Relative xpath for Email edit box with multiple attribute values

//input[@name='identifier'][@type='email']

==================================================================================================================================================================

iii. \*\* Using AND & OR operators in xpath

AND: When given 2 attribute values are matches then only WD is able to identify web element

---

Syntax: //tagname[@attribute1='value' and @attribute2='value']

Ex: //input[@name='identifier' and @type='email']

Both should match

OR: It is used to identify webelement by using either one of the given attribute value can be Start/Stop

---

//\*[@name='Start' or @name='Stop']

==================================================================================================================================================================

\*\*iv. Using text():

============

Sometimes we can also prefer visible text of web element

Note: \*\* If the locator is dynamic never use text method , as value would be changing and you will be getting error.

If the value is static then we can go with this text() method

Ex: Find Relative xpath for "Create Account" web element in GMail login page

//span[text()='Create account']

Ex: Find Relative xpath for "forgotten account?" link in FB-homepage

//a[text()='Forgotten account?']

==================================================================================================================================================================

\*\*v. xpath with contains()method

===========================

To identify webelement based on some of the value matches with webelement attribute value or visibletext

Syntax:

//tagname[contains(@attribute,'value')] //if we use any attribute value

(or)

//tagname[contains(text(),'value')] //if we use visible text of webelement

==================================================================================================================================================================

Ex: Write script to click on "Forgotten account?" link in FB-homepage using contains() method in xpath

Script:

------

driver.findElement(By.xpath("//a[contains(text(), 'Forgotten')]")).click();

(or)

driver.findElement(By.xpath("//a[contains(text(), 'account?')]")).click();

==================================================================================================================================================================

vi. xpath with starts with() method

===============================

Using this method WD is able to identify webelement based on starting value of visibletext/attibutes

Syntax: //tagname[starts-with(@attribute-name,'value')]

(or)\

//tagname[starts-with(text(),'value')]

Ex: To click on "Forgotten account?" link

driver.findElement(By.xpath("//a[starts-with(text(),'Forgotten')]")).click();

==================================================================================================================================================================

vii. \*\* traversing xpath

================

It is a combination of relative xpath & absolute xpath

In following scenarios we prefer traversing xpath

a. When there are similar objects exist in a page

b. When webelement properties are changing during runtime

\*\*IQ (i.e. Dynamic objects)

c. When webelement doesn't consist any attribute value

==================================================================================================================================================================

Ex: When traversing xpath for a cell value in a web table

//table[@id='customers']/tbody/tr[2]/td[3]

Pageurl: https://www.w3schools.com/html/html.tables.asp

---> These are dynamic tables/objects will be changing

==================================================================================================================================================================

Ex: Find traversing xpath for "New Password" editbox with reference of "Create Account" section (i.e. Parent Object)

Parent-Child

============

---> Mouse over on the dynamic section of FB page to identify parent

//div[@id='reg\_form\_box']/div[4]/div/div/div

Ex: Find traversing xpath for "Next" button based on Google-Login section in GMail Login page

//div[@class='akfVF']/div/div[2]/div/div/div[2]/div/div[2]/div/div/div/span/span

=========================================================================END OF CLASS==========================================================================