id ✓

name ✓

link Text ✓

partialLinkText ✓

xpath

cssSelector }

[tagName

className]

tag Names $\longrightarrow$ <u>16</u> $\longrightarrow$ 0 - 1<u>5</u>

Find Element $\longrightarrow$ single

find Elements — multiple tags

Class Name →

Navigation Methods → Assignment

XPath Locator :-

HyperText
HTML
↳ tags

predefined → tags
↳
Web page creation

Xtensible Markup lang
XML
↓
Tags
↳
user defined
↳
Data Storage

X path → Absolute path → start from root

↓ XML

Relative path → Specific to element

Absolute path
/html/body/div[1]/div[1]/div[1]/div/div/div/div[2]/div/div[1]/form/div[1]/div[1]/input

//*[@id="email"] → Relative path

Relative Path :-

$\overset{TN}{\uparrow}$ $\overset{\text{A.Value}}{\nearrow}$ $\overset{\text{A.Value}}{\nearrow}$

< Input type = 'text' name = 'username' >

Syntax :- $\underset{A}{\downarrow}$ $\underset{A}{\downarrow}$

Class = 'abc def'

#1

// tagName [@attr_name = 'attrval']

//input [@class = 'abc def']

// input [@type = 'text']

//input [@name = 'username']

way 2 → [.]agroop (kaurr)

< input   name = user name' class =
↓ ↑A   user-abc A ✓
TN   user.abc ↗
    ' input text   abcd'>
                    A ✓

get the
→ value of attribute partially

//input[contains(@class, 'inputtext')]

//input[contains (@attr-name, 'partialvalue')]
//input[contains (@class, 'inputtext')]

Way 3 :- ⟨h2⟩ Heading ⟨/h2⟩

```html
<input type="text" class="inputtext _55r1 _6luy" name="email" id="email" data-
testid="royal_email" placeholder="Email address or phone number" autofocus="1" aria-
label="Email address or phone number">
```

```html
<span> Meta © 2022</span>
```
→ paired tag → Starting & Closing tags

```html
<h2 class="_8eso">Facebook helps you connect and share with the people in your life.</h2>
```

A.

↳ Content

`<span> Meta © 2022</span>`

exact as it is use

Syntax:-

// tagName [text () = ' exact value/content b/w two tags]

Eg.   // Span [text() = 'Meta © 2022']

1. `<a href="/web/index.php/auth/logout" role="menuitem" class="oxd-userdropdown-link">Logout</a>`

→ Locators :- Link Text, partialLinkText, className, TagName, , xpath, cssSelector

→ xpath :- //a[text()='Logout'] → exact content

[
//a[@href='/web/index.php/auth/logout'] — exact attr value

//a[contains(@href, 'logout')] — part attr vl.

//a[contains(text(),'Log')] ↪ partial content

{ tagName A ① ↑ AV ② ↑ AV ↑ A

<h5 class="oxd-text oxd-text--h5 orangehrm-login-title" data-v-7588b244="" data-v-30b9e6c4="">Login</h5>

A AV

atl = ' '

→ TagName → _tagName //h5[contains(text(), 'Login')]

→ ClassName

→ xPath //h5[text() = 'Login']

→ x cssSelector //h5[@class = 'orangehrm-login-title'] ← exactvalue

//h5[contains(@class, 'orangehrm-login-title')]

#Way 4: // tagName[contains (text() , 'value')]
  or
  partial Value

`<a>. Link </a>`

`<span>  Testing   Demo    </span>`

`<h5>  Heading5  Value  </h5>`

// h5[contains (text() , ' Heading5')]

→ //a [text()= '. Link . ']   → exact

→ //a [contains (text() , 'Link')]

→ //span[contains (text() , 'Testing Demo')]

CssSelector: Locator

<input type = 'text' id = 'abc']

Class - .

id   -#

Xpath : //input [@id = 'abc']

CssSelector -   #abc - id

1. driver. findElement (By. cssSelector ('#abc'))

2. input [id = 'abc']

1. input#email → tagName with 'id' attribute value

2. input[name='email'] → tagname with 'name' attribute value

3. #email → # id attr value

4. input[type='text'] → css Selector with attribute value

   Syntax :- tagName[attr_name = 'attr val']

CSS Selector with class Name

Class

1. ._8eso → only Class Name

2. h2._8eso → tag Name with class Name
   .class

3. h2[class='_8eso'] → tag Name with class Name attribute