**WebDriver Locator Techniques:**

**Locators supported by Selenium Webdriver:**

* ID
* Name
* ClassName
* linkText
* partialLinkText
* tagName
* Xpath
* CssSelector

**What are we going to learn in the next few chapters?**

Finding an element by ID (driver.findElement(By.id("firstname"))

Finding an element by Name (driver.findElement(By.name("firstname"))

Finding an element by className (driver.findElement(By.className("beautifultextbox"))

Finding an element by Link Text (driver.findElement(By.linkText("google the top search engine"))

Finding an element by Partial Link Text (driver.findElement(By.partialLinkText("google"))

Finding an element by xpath

Finding an element by CSSSelector

Finding elements when the ID, Name, attributes have spaces

* Find an element by id – works
* Finding an element by name – works
* Finding an element by Class Name – Produces error
* Finding element by xpath – works
* Finding element by CSS Selector - works

Finding an element by Xpath

* Writing our own xpaths ("//input[@color='red']"))
* Writing xpaths with regular expression (By.xpath("//\*[@id='firstname']"))
* Writing xpaths by using contains keyword (By.xpath("//input[contains(@id,'last')]"))
* Writing xpaths with starts-with keyword (By.xpath("//input[starts-with(@id,'last')]"))
* Writing xpaths with ends-with keyword (By.xpath("//input[ends-with(@id,'name')]"))
* Finding an element using parent child relationship (By.xpath("//div[@name='pcrel']/div/div/input"))
* Finding siblings (By.xpath("//li[@id='list1']/following-sibling::li[2]"))
* Finding parent elements (By.xpath("//div[@id='child2']/parent::div"))
* Finding ancestors (By.xpath("//div[@id='child2']/ancestor::div")

Finding an element by CSS Selector

* Writing our own CSS Selectors using single attribute (By.cssSelector("input[color='red']"))
* Writing our own CSS Selectors using multiple attributes (By.cssSelector("input[color='red'][type='text']"))
* Other ways of writing css selectors tagname#id (By.cssSelector("input#firstname"))
* Other ways of writing css selectors #id (By.cssSelector("#firstname"))
* Other ways of writing css selectors tagname.className (By.cssSelector("input.beautifultextbox"))
* Other ways of writing css selectors tagname.className (By.cssSelector("input.beautifultextbox.colorful"))
* Other ways of writing css selectors tagname.className (By.cssSelector(".beautifultextbox"))
* Other ways of writing css selectors tagname.className (By.cssSelector(".beautifultextbox.colorful"))
* CSS Locators for Sub String Matches – Starting text match with symbol ^ (By.cssSelector("input[color^='red']"))
* CSS Locators for Sub String Matches – Ending text match with symbol $ (By.cssSelector("input[color$='red']"))
* CSS Locators for Sub String Matches – Anywhere text match with symbol \* (By.cssSelector("input[color\*='red']"))
* CSS Locators using Child Locators (By.cssSelector("form>label>input[id='firstname']"))
* When you find more than one element (findElements(By.cssSelector("input[class\*='text']")).get(2))

Finding first element using tag name and enter the value in first element driver.findElement(By.tagName("input")).sendKeys("Venkat");

Finding all elements using tag name and enter the value in a specific element driver.findElements(By.tagName("input")).get(2).sendKeys("Venkat");

Finding an element by visible text ("//div[contains(text(),'I am child 2')]"))