|  |  |
| --- | --- |
| **List some of the Locators?** |  |
| 1. Id: Select element with the specified @id attribute. 2. Name: Select first element with the specified @name attribute. 3. Linktext: Select link (anchor tag) element which contains text matching the specified link text 4. Partial Linktext Select link (anchor tag) element which contains text matching the specified partial link text 5. Tag Name Locate Element using a Tag Name. 6. Class name Locate Element using a Class Name. 7. Css Select the element using css selectors. 8. Xpath Locate an element using an XPath expression. |  |
| **How to find element using id locator?** |  |
| WebElement searchbox= driver.findElement(By.id("….")); |  |
| **How to find element using name locator?** |  |
| WebElement searchbox= driver.findElement(By.name("….")); |  |
| **How to find element using LinkText locator?** |  |
| WebElement loginLink= driver.findElement(By.linkText("…..")); |  |
| **How to find element using Partial LinkText locator?** |  |
| WebElement loginLink= driver.findElement(By.partialLinkText ("…..")); |  |
| **How to find element using Class locator?** |  |
| WebEment theme = driver.findElement(By.className("….")); |  |
| **How to find element using TagName locator?** |  |
| List<WebElement> myElementListByTag = driver.findElements(By.tagName("…")); |  |
| What is the standard of XPath |  |
| Syntax:  //tag[@attribute='value'] |  |
| **How to use CSS (Cascading Style Sheets)?** |  |
| Syntax:  tag[attribute<special character>=’value’]   |  | | --- | | Alternative Symbols of id:   * “#” -> Id * Example: input[id=displayed-text] can be written as  1. input#displayed-text 2. #displayed-text | | Alternative Symbols of class:   * “.” -> Class * Example: input[class=displayed-class] can be written as  1. input.displayed-class 2. .displayed-class  * We can also write : .class1.class2.class3 -> Until we find a unique element * .inputs.displayed-class | | Using special character/wildcards in CSS Selectors:   1. “^” -> Represents the starting text 2. “$” -> Represents the ending text 3. “\*” -> Represents the text contained   Examples:  input[class^='inputs'] -> Two matching nodes  input[class$='class'] -> One matching node  input[class='displayed-class'] - No matching nodes  input[class\*='displayed-class'] -> One matching node | | Finding Children: using ‘>’ symbol  Examples:  fieldset>table  fieldset>#product -> One matching node  fieldset>button -> One matching node  fieldset>a  fieldset>input#name |  |  | | --- | | Details of Symbols: | | **Alternative Symbols of id**:   * “#” -> Id * Example: input[id=displayed-text] can be written as  1. input#displayed-text 2. #displayed-text | | **Alternative Symbols of class**:   * “.” -> Class * Example: input[class=displayed-class] can be written as      1. input.displayed-class 2. .displayed-class  * We can also write : .class1.class2.class3 -> Until we find a unique element * .inputs.displayed-class | | Using special character/wildcards in CSS Selectors:   1. “^” -> Represents the starting text 2. “$” -> Represents the ending text 3. “\*” -> Represents the text contained   Examples:    input[class^='inputs'] -> Two matching nodes  input[class$='class'] -> One matching node    input[class='displayed-class'] - No matching nodes  input[class\*='displayed-class'] -> One matching node | | Finding Children: using ‘>’ symbol.  Examples:    fieldset>table  fieldset>#product -> One matching node    fieldset>button -> One matching node    fieldset>a    fieldset>input#name | |  |
| How to use XPath? |  |
| Syntax:  //tag[@attribute='value']       |  | | --- | | <https://letskodeit.teachable.com/p/practice> | | 1. To get the second child element ( /li[2] ):     //div[@id='navbar']/div/div/div/ul**/li[2]/**a | | 1. Use of standard XPath syntax (**a[@class='navbar-link fedora-navbar-link']**):   //div[@id='navbar']//ul/li[2]/**a[@class='navbar-link fedora-navbar-link']** | | 1. Use of : 2. **contains** : //tag[contains(@attribute,'value')]   instead of //tag[@attribute='value']   1. **and** : tag[condition1 and condition1]//tag   //div[@class='navbar-header']//a**[contains(@class,'navbar-link') and contains(@href,'sign\_in')]** | | 1. use of starts-with: 2. **contains** : //tag[starts-with(@attribute,'value')]   instead of //tag[@attribute='value']  //div[@class='navbar-header']**//a[starts-with(@class,'fedora')]** | | 1. Use of text() alone : 2. **contains** : //tag[text()='value')]   instead of //tag[@attribute='value']  //div[@class='homepage-hero']//**a[text()='Enroll now']** | | 1. Use of text() with contains : 2. **contains** : //tag[contains(text(),'value')]   instead of //tag[@attribute='value']  //div[@class='navbar-header']//**a[contains(text(),'Log')]** | | 1. XPath standard will work on href as well:   //a**[@href='/sign\_in']** | |  | | 1. Use of 2. Parent:: 3. Preceding-sibling:: 4. Following-sibling::   //a[@href='/sign\_in']//**parent::**li//**preceding-sibling::**li//**following-sibling::**li[1] | | 1. Use of position():   //div[@class='datepicker-cal-month']**[position()=1]** | | 1. Use of and :   //div[@class='datepicker-cal-month'][position()=1]//button**[@class='datepicker-cal-date' and contains(text(),'22')]** | |  |