

## **2. Locating Web Page Elements**

### **1. Using ID as a Locator**

- Open Eclipse
- Finding Web element using Locator **ID**
  - a. Syntax : id = id of the element
  - b. Example : `driver.findElement(By.id("Email"));`

### **2. Using class name as a Locator**

- Finding Web element using Locator **ClassName**
  - a. Syntax : class = Class Name of the element
  - b. Example : `driver.findElement(By.class("classname"));`

### **3. Using Name as a Locator**

- Finding Web element using Locator **Name**
  - a. Syntax : name = Name of the element
  - b. Example : `driver.findElement(By.name("name"));`

### **4. Using LinkText as a Locator**

- Finding Web element using Locator **Link Text**
  - a. Syntax : link = partialLink of the element
  - b. Example : `driver.findElement(By.partialLinkText("plink"));`

### **5. Using Xpath as a Locator**

- Finding Web element using Locator **Xpath**
- Xpath can be created in two ways
  - a. **Relative Xpath**
    - Syntax : relativeXpath : `//*[@class='relativexapath']`
    - Example : `driver.findElement(By.xpath("//*[@class='relativexapath']"));`
  - b. **Absolute Xpath**
    - Syntax : absoluteXpath : `html/body/div[1]/div[1]/div/h4[1]/b`
    - Example :  
`driver.findElement(By.xpath("html/body/div[1]/div[1]/div/h4[1]/b"));`

## 6. Using Xpath as a CSS Selector

- CSS Selector have many formats, namely
  - a. **Tag and ID**
    - Syntax : "css = tag#id"
    - Example : `driver.findElement(By.cssSelector("input#email"));`
  - b. **Tag and Class**
    - Syntax : "css = tag.class"
    - Example : `driver.findElement(By.cssSelector("input.inputtext"));`
  - c. **Tag and Attribute**
    - Syntax : "css = tag[attribute=value]"
    - Example :  
`driver.findElement(By.cssSelector("input[name=lastName]"));`
  - d. **Tag, Class and Attribute**
    - Syntax : "tag.class[attribute=value]"
    - Example :  
`driver.findElement(By.cssSelector("input.inputtext[tabindex=1]"));`
  - e. **Inner text**
    - Syntax : "css = tag.contains("innertext")"
    - Example : `driver.findElement(By.cssSelector("font:contains('Boston')"));`

## 7. Using Xpath Handling complex and Dynamic elements

- Dynamic Xpath has many formats, Namely
  - a. **Contains();**
    - Syntax : "xpath = //\*[contains(text(),'text')]"
    - Example : `driver.findElement(By.xpath("//*[@contains(text(),'sub')]"));`
  - b. **Using OR & AND**
    - Syntax : `xpath=//*[@type='submit' or @name='btnReset']`
    - Example :  
`driver.findElement (By.xpath("//*[@type='submit' or @name='btnReset']"));`
  - c. **Start-with function**
    - Syntax : `xpath= //label[starts-with(@id,'message')]`
    - Example :  
`driver.findElement (By.xpath("//label[starts-with(@id,'message')]"));`
  - d. **Text();**
    - Syntax : `xpath=//td[text()='UserID']`
    - Example : : `driver.findElement (By.xpath("//td[text()='UserID']"));`

**e. Following**

- Syntax : `xpath=//*[@type='text']//following::input`
- Example :  
`driver.findElement(By.xpath("//*[@type='text']//following::input"));`

**f. Preceding**

- Syntax : `xpath=//*[@type='text']//preceding::input`
- Example :  
`driver.findElement(By.xpath("//*[@type='text']//preceding::input"));`

**g. Following - sibling**

- Syntax : `xpath=//*[@type='submit']//preceding::input`
- Example :  
`driver.findElement (By.xpath ("//*[@type='text']//following-sibling::input"));`

**Pushing the code to your GitHub repositories : -**

- Open your folder where the Project . And then click the right button to open the git bash command prompt.
- Before that, open the github and create a new repository.
- Initialize your repository using the following command:

`git init`

- Add all the files to your git repository using the following command:

`git add .`

- To check the status of the repository use the below command:

`git status`

- Commit the changes using the following command:

`git commit . -m "Changes have been committed."`

- To add the files to the repository use the (URL) from the github and use the command;

`git remote add origin <url>`

- Push the files to the folder you initially created using the following command:

`git push origin master.`