**Q. What is Locators?**

* Locators are used to locate certain WebElement.
* Locators are set of methods coming from Selenium jar files/library.
* There are 8 different locators coming from By class.
* There are new functions to locate nearby elements by specifying directions with SE 4.  
     
  1)  ID : It is always unique and most preferred locator if available.  
  2)  Name : It is used when unique name available.  
  3)  TagName : It is recommended to use with findElements() method.  
  4)  ClassName : Not recommended. Usually not unique.  
  5) LinkText : Can be used only with links. We must give entire text.  
  6) PartialLinkText      : Can be used only with links. We can give part of text.  
  7)  Xpath : Mostly used locators after ID (Dynamic)  
  8)  CSS Selector  : Mostly used locators after ID. It is faster than Xpath (Dynamic)

**Q. What is XPATH?**         Address of the web element.                                          
- One of the most used locator to find elements after ID.  
- Used to navigate the DOM (Document Object Model) elements.  
- We can write XPath for any Web element  
- We must use indexes if there are more than one element.  
- We can use \* and . to find any element, any text etc.…

**Q. Different types of Xpath:**

**There are two types of Xpath. Absolute Xpath and Relative Xpath.**

1. **Absolute Xpath:**

* /html/body/div/header/div/div/div
* It is the direct way to find the element, but the disadvantage of the absolute XPath is that if there are any changes made in the path of the element then that XPath gets failed.
* Not recommended because we go from the first parent to target child one by one.
* It can easily be broken when a new element is added on the page.
* It begins with a single forward-slash (/).

1. **Relative Xpath:**

* //tagname[@attribute=’value’] or //\*[.=text name];
* It starts from the middle of HTML DOM structure. It starts with double forward slash (//).
* It can search elements anywhere on the webpage, means no need to write a long Xpath and you can start from the middle of HTML DOM structure.
* Relative Xpath is always preferred as it is not a complete path from the root element.

**Q. Difference between XPATH and CSS Selector?**

An important difference between CSS and XPath locators in Selenium is that **CSS looks for elements going down the DOM**, while **XPath allows you to navigate both up and down**. This means that using XPath, you can find child web elements and then easily capture their parent or other ancestor.

**🡺 XPATH**

* XPath can jump from parent to child so that we can locate specific web element.
* Xpath navigates through the DOM up and down.
* Xpath allows to handle dynamic elements (contains (), starts-with (), text (), parent ()).
* Xpath has different syntax (// is used in xpath but not in CSS).
* If XPath returns multiple elements we use index but not reliable.  
  ===> //tag[@attribute=‘value’]  
  ===> //\*[.=‘Hello World’] à This will returns all of the Hello World text elements in the page  
  ===> To find an element by EXACT TEXT you can

//\*[.='text name'] OR //\*[(text()='text name’)] OR  //\*[contains(text(),’Piece Of Text’)]

* XPath engines are different in each browser, making them inconsistent

**🡺 CSS Selector:**

* **Tag and ID**
* **Tag and Class**
* **Tag and attribute**
* **Tag, Class and attribute**
* CSS Selector is a little faster and better than Xpath.
* CSS Selector locates without navigation.
* CSS Selector usually same across the browsers so it is more reliable.
* CSS Selector has no way to select the parent of an element
* CSS Selector can almost do everything what XPath does  
  o  [#idValue](https://www.linkedin.com/feed/hashtag/?keywords=idvalue&highlightedUpdateUrns=urn%3Ali%3Aactivity%3A6890713187327492096)  
  o  .className  
  o  [attr-name^=’value’] à Instead starts-with  
  o  [attr-name$=’value’] à Instead ends-with  
  o  [attr-name\*=’value’] à Instead contains

