JAVA/SELENIUM : Testing Framework

What's new in Selenium 4

LevelUp360 : anshulc55@icloud.com

Java/Selenium Testing Framework Automation

- Selenium 4 : Selenium 4 alpha 3 released on April 2019.
  - Selenium 4 have few major improvements over Selenium 3.
  - No Major changes in Selenium 3 generic functionality or end user usability.

- Few Major features are introduced in Selenium 4.

Java/Selenium Testing Framework Automation

- **Enhanced version of Selenium Grid -**
  - Earlier version of Selenium Grid was complex to set up and rigid in terms of scaling.
  - New Selenium Grid comes with Docker support. This will enable developers or testers to spin up the containers rather than setting up heavy virtual machines.
  - Grid in Selenium 4 also comes with an enhanced user-friendly GUI.

Java/Selenium Testing Framework Automation

- **Upgraded Selenium IDE -**
  - Improved GUI for intuitive user experience.
  - Improved control flow mechanism that enables testers to write better “while” and “if” conditions.
  - The code for test cases recorded using Selenium IDE can be exported in the desired language binding like Java, C#, Python, .NET, and JavaScript.

- **Relative Locators in Selenium 4 -**
  - Selenium 4 brings an easy way of locating elements with the inclusion of relative locators.
    - To Left Of, To Right Of, Above, Below

- **Support for Chrome Debugging Protocol -**
  - Selenium 4 comes with native support for Chrome DevTools Protocol. This means QAs can now use Chrome development properties like Fetch, Network, Profiler, Performance, Application cache, and more.

- **Better Window/Tab Management in Selenium 4 -**
  - **newWindow** that allows users to create and switch to a new window/tab without creating a new WebDriver object.

```
driver.switchTo().newWindow(WindowType.WINDOW);

driver.switchTo().newWindow(WindowType.TAB);
```

- **Modifications in the Actions Class -**
  - **click(WebElement)** - This method is added to Actions class to replace the *moveToElement(onElement).click()*. It is used to click on a certain web element.
  - **clickAndHold(WebElement)** - This method will replace the *moveToElement(onElement).clickAndHold()*. It is used to click on an element without releasing the click.
  - **contextClick(WebElement)** - This method will replace *moveToElement(onElement).contextClick()*. It will perform the right click operation.
  - **doubleClick(WebElement)** - This method is added to replace *moveToElement(element).doubleClick()*. It will perform a double click on an element.

Thank You...

Don’t be the Same! Be Better!!!

LevelUp360 : anshulc55@icloud.com