Chapter 2

Aced Selenium in Real Time

Which of the webdriver APIs is the fastest and why?

HTML UnitDriver is the fastest of all as it doesn't execute in browser. It employs a simple https request - response mechanism for test cases execution.

What is the super Interface of Webdriver interface?

SearchContext

What is the syntax for CSS Selector in selenium?

* by id: css=tag#id
* by class: css=tag.classname
* by attribute: css=tag[attribute=value]
* by substring: css=tag[attribute^='prefix of string'] css=tag[attribute$='prefix of string']
* by inner text: css=tag:contains(text)

How to get css property of the webelement in selenium?

* driver.findelement(By.id("id")).getcssvalue("nameofattribute");

How to move to nth child element?

* Using xpath - div[n]
* Using css- div:nth-child(2)

What is Synchronization?

* It is a mechanism that involves 2 or more components that work parallel with each other.
* In test automation we have 2 components: Application under test, Test automation tool
* Both Components have their own speed and to make these 2 components move with same and desired speed, we should write our scripts in such a way that we will not encounter "element not found" errors.
* we have 2 categories of Synchronization - Unconditional, conditional
* In Unconditional Synchronization we specify timeout value only, the tool will wait until a certain amount of time and then proceed further.
* Examples are wait(), thread.sleep().
* Disadvantage of this is that even though the application is ready there is chance of unnecessary waiting time.
* In Conditional synchronization we specify a condition along with timeout values, so the tool waits and checks for the condition and then it come out if nothing happens.
* It is important to set timeout value in conditional synchronization as the tool should proceed further instead of making the tool to wait for a particular condition to satisfy.
* Examples are implicit wait, explicit wait.

What is FluentWait in selenium?

* It is used to define maximum time for the webdriver to wait for condition as well as frequency with which we want to check the conditions before throwing an exception.
* It checks for the webelement at regular intervals until the object is found or timeout happens.
* Wait wait = new FluenWait(driver).withTimeout(timeout,seconds).pollingEvery(timeout, seconds).ignoring(expection.class);
* Above is deprecated in selenium v3.11 and above
* Wait wait = new FluentWait(driver).withtimeout(30, Timeunit.seconds).pollingEvery(5,Timeunit.seconds).ignoring(NosuchElementException.class)

What is default polling time for explicit wait and implicit wait?

* 500 milliseconds for explicit wait.
* Implicit wait doesn't have any polling time.
* It waits for complete specified time so default polling time is 0.

What is difference between setsleep() and sleep()?

* Both are used to delay execution speed, setspeed setup speed that will apply a deplay time before every selenium operation setspeed("5000") - it waits for 5 seconds.
* Sleep() setup wait only for once when called in our selenium script sleep(5000) - it waits for 5 seconds.

How to make sure page is loaded using selenium?

Text

Description automatically generated

What is throwable?

* It is a parent class for error and exception.

What are causes of staleElementException?

* The referenced web element has been deleted completely.
* The referenced element is no longer attached to DOM

How to handle staleElementException?

* Refreshing the web page using navigate().refresh and then perform action on the webelement
* Using try catch block in the for loop with retry count or while loop condition
* Using expected conditions.refreshed in explicit wait condition (wait.until(expectedConditions.refreshed(Expectedconditions.stalenessOf("")))

How to throw custom exception?

Text

Description automatically generated with medium confidence

Can we set the webdriver to static?

web driver if set to static works fine and can be used in multiple classes with the single instance effectively. But when we do the parallel execution, the instance from both the threads try to modify the same browser and will create a problem. So it is recommended to keep webdriver non static

What is same-origin poliy? How can we avoid it?

* It is introduced for security reasons
* It ensures that the content of our site will never be accessible by a script from another site.
* As per policy, any code loaded with browser can only operate within that websites domain..
* To avoid this same -origin policy, the proxy injection method is used
* In the proxy injection mode, selenium server tricks the browsers to be a real HTTP URL ,ie, it acts as a client-configured HTTP proxy, which sits between the browser and the application under test.(AUT) and then marks the AUT under a fictional URL.

A picture containing text, whiteboard

Description automatically generated

How many parameters can selenium commands have at minimum?

* 4 parameters - host, url, browser, port number

How to redirect browsing from a browser through same proxy?

* We have PROXY class to redirect browsing from a proxy.

Text

Description automatically generated

How to handle https website in selenium?

Graphical user interface, text, application

Description automatically generated

How to login to any site if it is showing any authentication popup?

* http:\\username:password@url

How to launch a batch file in selenium webdriver?

A screenshot of a computer

Description automatically generated with medium confidence

How to scroll down page using Javascriptexecutor?



How to scroll to particular element?

Text

Description automatically generated

How to send text using JavaScript executor?

Text

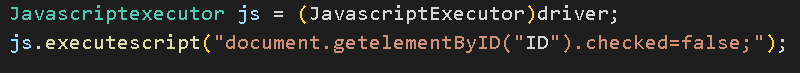
Description automatically generated

How to click button using JavaScript executor?

Text

Description automatically generated

How to click on checkbox using JavaScript executor?



How to generate alert popup using JavaScript executor?

Text

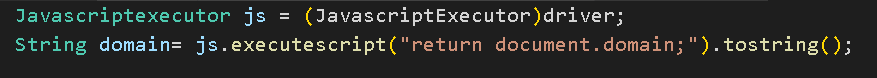
Description automatically generated

How to refresh browser window using JavaScript executor?

Graphical user interface, text

Description automatically generated

How to get domain in selenium using JavaScript executor?



How to get title in selenium using JavaScript executor?

Text

Description automatically generated

How to get URL of webpage in selenium using JavaScript executor?

Text

Description automatically generated

How to navigate to URL in selenium using JavaScript executor?

Text

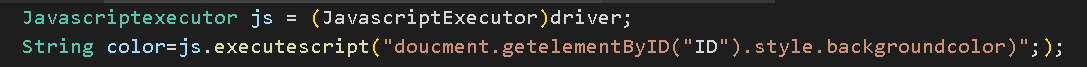
Description automatically generated

How to get the hidden element in selenium using JavaScript executor?

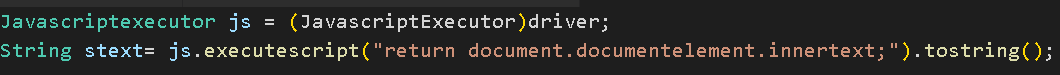
Graphical user interface, text

Description automatically generated

How to get background color of element in selenium?



How to get inner text in selenium using JavaScript Executor?



How to send enter/tab keys in web driver?

Text

Description automatically generated with medium confidence

How to type text in a new line inside a text area?



How to press shift + tab?

A picture containing text

Description automatically generated

How to enter Shift/ALT/Control Keys?

Text

Description automatically generated

What are different network protocols that selenium supports?

* http protocol
* No authentication - driver.get("http:\\url");
* Basic authentication - webdriver.get("https:\\"+username+":"+password+"@"+"url);
* Using Preferences:
* FirefoxProfile ff= new FirefoxProfile();
* ff.setPreference("network.http.phisy-userpass-length",255);
* driver = new Firefoxdriver(FF);
* driver.get("http:\\user:password@url");

How to handle SSL for Firefox in selenium?

Text

Description automatically generated

How to handle SSL for Chrome in selenium?

* Chrome had to set SSL options via desired capabilities

Text

Description automatically generated

How to handle SSL for Internet Explorer in selenium?

Text

Description automatically generated

How to identify tooltip by using web driver?

* We can test this by validating the text of the object as text of the object is used as tooltip text "title" of the object = tooltiptext
* StaticToolTip- Selenium can capture the static tooltip by calling getattribute('title') method.
* Dynamic Tooltip - This is usually created by iquery/javascript plugins. Using actions class we can get HTML(div tag) once we have mouse hover link.

Text

Description automatically generated

How to get all the values from table?

Text

Description automatically generated

How to take screenshot of web element?

* Ashot is third party utility by Yandex supported by selenium Webdriver to capture the screenshot.

A screenshot of a computer

Description automatically generated

How to click on each link and navigate back?

Text

Description automatically generated

How to count the number of links in a page?

Text

Description automatically generated with low confidence

How to get all the links on a webpage?

Text

Description automatically generated

How to find broken links on a webpage?

* First collect all links on page based on tag
* Then send an http request for the link and read the http response code
* Based on the response code determine whether the link is valid or broken

Text

Description automatically generated

How are method overloading and method overriding implemented in selenium?

method overloading in selenium:

* implicit wait in selenium
* driver.manage().timeout().implicitwait(10,timeunit.seconds)
* driver.manage().timeout().implicitwait(10,timeunit.minutes)
* driver.manage().timeout().implicitwait(10,timeunit.hours)
* frames in selenium:
* using index: driver.switchto.frame();
* using framename: driver.switchto().frame(name)
* using webelement: driver.switchto().frame(webelement)

Method overriding in selenium:

* get and navigate methods, both are used for navigating to the url of webelement.

How to validate if the element is highlighted?

* Get the color of the element using getCssvalue("color")
* Now compare the value with the desired value, if both are equal the element is highlighted.

How to handle slider in selenium/ draganddrop in selenium?

* DragandDrop method can be used to drag the slider
* DragandDrop method with 2 parameters - Actions.dragAndDrop(Sourcelocator, Destinationlocator)
* DragandDrop method with 3 parameters - Actions.dragAndDropBy(Sourcelocator, x-axis pixel of Destinationlocator, y-axis pixel of Destinationlocator)

Text

Description automatically generated

How to enter text in selenium without using sendkeys()?

* By using javascriptexecutor: js.executeScript("arguments[0].setAttribute('value',"texttoenter", webelement)

How to handle ajax calls using selenium webdriver?

* AJAX means asynchronous javascript and xml, it allows the web page to retrieve small amounts of data from the server without reloading the entire page.
* AJAX sends http requests from the client to server and then process the server response without reloading the entire page, so normal wait commands will not work as the actual page is not refreshed.
* The best way is to use the dynamic waits like in explicit wait adding condition as wait.until(expectedconditions.elementtobeclickable), wait.until(ExpectedConditions.alertpresent), wait.unitl(expectedconditions.texttobepresentinelement)

What does build(), perform() methods mean in actions class?

* Build() method is used to create chain of actions or operations we want to perform
* perform() is used to execute chain of actions which are build using build()
* whenever using multiple actions always use build().perform() as it creates chain and excute them , if using only one action , can use perform() directly

What are the OOPs concepts implemented in Selenium Automation Framework?

* Abstraction: It provides us to hide the implementation of internal details and show the functionality to user. Example: in POM pattern, we write locators and the methods in page class and we utilize those those locators in the tests but we can't see implementation of the methods so here we are hiding the implementation of locators from the tests. In java abstraction is achieved by abstract class and interfaces.
* Interface: Webdriver driver = new Chromedriver() , here webdriver is the interface and chromedriver is the class implementing the interface.
* Inheritance: Once class acquiring the properties of anther class is called inheritance and in automation, we create a base class to initialize the webdriver interface, webdriver waits, property files, excels etc and throughout the framework we extend this base class in other classes like test and utility classes.
* Polymorphism: Method Overloading: implicit wait in selenium has different time stamps such as seconds, minutes, hours etc. Also, action class and assert class are also examples of overloading
* Method Overriding: get and navigate methods of different drivers n selenium
* Encapsulation: Mechanism of binding code and data (variables) together in a single unit. all the classes in the framework are examples of encapsulation. Also, in POM class we declare the data members using @FindBy and initialization of data members will be done using constructors to utilize those in methods

What are challenges faced in selenium while automating in the project?

You can put over all the challenges faced while automating your own project, I am putting here few of the challenges I faced in my experience.

* The first challenge is that as we know selenium can automate only web-based applications, when I got the scenario to automate the application where I need to download the file and check in windows file explorer and unzipping it and opening file in the windows software and editing it and saving and zipping it. It needs lot of research and able to automate using the java inbuild libraries to handle windows explorer related validations and using autoIt tool for automating the scenarios related to validation of windows software
* One of the other scenarios is getting the history of the browser deleted before running the scenarios without using delete cookies and going the chrome settings and deleting the history
* While performing cross browser testing, many of the scenarios which passed in chrome failed in IE browser as the locators are not identified in the IE browser
* In the BDD Cucumber framework, implementation of skipping the failed steps and moving to the next step in the same scenario let me explore more and able to do it using plugins
* Also rerunning the failed scenarios without using testNG retry mechanism is also bit challenging