**Automation Testing**: -

It is nothing but to test the s/w automatically with the help of some programming languages and with the help of tools.

**WHY Selenium**: -

1.It is a open source automation tool and it is used to test web applications only.

2.It supports with many programming languages ex-java, python, c#.

3.It works multiple browsers—chrome, Firefox, edge

4.It is compatible with different operating systems like MAC, Windows, Linux.

**Selenium flavours**: -

**Selenium IDE**

It is the (selenium integrated development environment) simplest tool.

It works based on RECORD and PLAYBACK functionality. That is it stores the user interactions and exports them for code reusability script.

**Selenium RC**-- It is remote control.

**Selenium WebDriver**-

It is a web framework used to perform cross browser testing and automate the application whether it’s working expectedly or not. We can choose the programming language like java, ruby, c # etc.

**Selenium Grid-**

Grid works combination of both selenium RC and web driver. It performs cross browser testing.

It can execute the test cases on multiple system parallelly as well.

Note: -

When we are automating with Firefox driver there is no need to write system. Set property. But whereas with chrome and internet explorer we must write setup.

Navigation Methods: -

Navigate methods works similar to get method. It’s navigated to the URL.

The main difference b/w get and navigate is it is not wait for page loading. It accepts both URL and string parameters.

It does not clear the cache memory.

Driver. Navigate (). Forward () -🡪 which move to the next URL.

Driver. Navigate (). Back () -🡪 which move to the back page.

Driver. Refresh () 🡪 to refresh the page .

**Close AND Quit: -**

Close () 🡪 it closes the current window.

Quit () -🡪 closes all windows that is entire browser.

**Radio Buttons**: -

We can automate the radio button by inspecting them.

If the button is selected, we can simply check with

Ele.is enabled/is selected.

If the button is selected, we can simply check with

Ele.is disabled.

Dynamic Xpath:-

We can write the dynamic Xpaths with different ways

1.starts-with

//\*[starts-with(text(),’value’)]-🡪starting of the value

2.contains

//\*[contains(text(),’value’)]

//\*[contains(@attribute=’value’]

//\*[ends-with(text(),’value’)]-🡪ending of the value

**Handling frames**: -

To handle the no of frames

1.Driver. switchTo (). Frame (select either by index or web element/name)

If the driver moves from sub frame to parent frame use

Driver. switchTo (). Default content ()

**Drag and Drop: -**

To dragging the frame we need to find the element by inspecting the particular item.

Web Element drag=driver. findelement (By.id(“value”);

Actions action=new Actions(driver);

Action. moveToElement (drag). drag And Drop(drag,50,50)

That means that drag that particular element to 50 x coordinates and 50 y coordinates.

For sliding the window

Actions action=new Actions(driver);

Action. moveToElement (drag). drag And Drop(slider,50,50)

**Encode and decode the passwords: -**

To avoid the hacking conditions, we need to encode the data. For encoding the data, we need to convert into Base64.

String a=” password”

String en=Base64.getEncoder(). encode (a. get bytes ()).

String de=Base64.getEncoder(). Decode (en. get bytes ()).