**Selenium Introduction**

\*Selenium is one of the most widely used **open source Web UI (User Interface) automation testing suite.**

\* It was developed by Jason Huggins in 2004

\*Supports automation across **different browsers, platforms and programming.**

\* Easily deployed on platforms such as **Windows, Linux, Solaris and Macintosh.**

\*Supports OS (Operating System) for mobile applications **like iOS, windows mobile and android.**

\* Languages supported by Selenium include **C#, Java, Perl, PHP, Python and Ruby.**

\* Browsers supported by Selenium include **Internet Explorer, Mozilla Firefox, Google Chrome and Safari.**

\* Selenium test scripts can be coded in any of the supported programming languages and can be run directly in most modern web browsers.

\* Selenium can be used to automate functional tests and can be integrated with automation test tools such as **Maven**, **Jenkins**,**& Docker to achieve continuous testing.**

\* Selenium can also be integrated with tools such as **for managing test cases and generating tool:- TestNG**, & **JUnit**

|  |  |  |  |
| --- | --- | --- | --- |
| **Test cases managing and generating tool** | **Defect/Bug tracking tools** | **Performance testing tools (Load testing tools)** | **Test Management Tools** |
| **TestNG**, & **JUnit** | **Jira, Bugzilla, BugNet, Redmine, Mantis, Trac, Backlog.** | **Apache JMeter, Load Runner[HP], Load Ninja, Web LOAD, LoadComplete, NeoLoad, LoadView.** | **Testlink**, **Quality center, RTH, Testpad, Test Monitor, PractiTest** |

# (best for test managing, scheduling, defect logging, tracking, and analysis.)

* **Selenium can be integrated with frameworks like Ant and Maven for source code compilation.**
* **Selenium can also be integrated with testing frameworks like TestNG for application testing and generating reports**.

\*\***Selenium Software is not just a single tool but a suite of software, each piece catering to different Selenium QA testing needs of an organization. Here is the list of tools**

* **Selenium Integrated Development Environment (IDE)**
* **Selenium Remote Control (RC)**
* **WebDriver**
* **Selenium Grid**

**\*\*\*\*\*\*\*\*\*\*\*SELENIUM- IT IS A OPEN SOURCE TOOL, USED TO TEST THE WEB BASED APPLICATION.**

* **Selenium RC + WebDriver = (New & Improved) Selenium RC**

**\*Automation Testing tools which are used for functional automation:-**

**-🡪QTP, Selenium, Coded UI, Auto IT, Rational Robot.**

**\*Automation Testing tools which are used for Non-functional automation:-**

* **-🡪Load Runner, JMeter , Burp Suite, Acunetix.**

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Sir Notes\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**->IF THE PROJECT IS REPETETIVE , WE CAN GO FOR AUTOMATION TESTING.**

**->ONE TIME REQUIREMENT PROJECT (If the project tested at 1 time ),THEN WE CAN USE MANUAL TESTING.**

**SELENIUM ---🡪 (mediator)CHROME DRIVER-----🡪 GOOGLE CHROME**

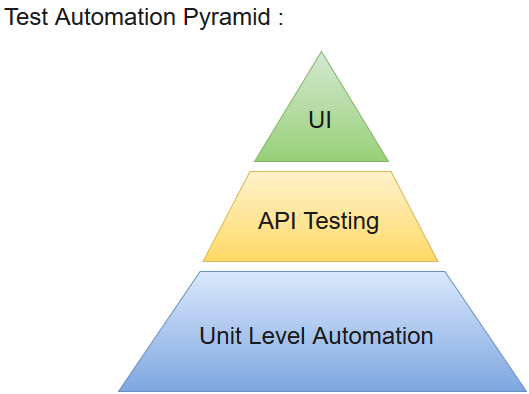
**Locator-To find the location/ address of that Web element.**

**\*Locators:-**

**1. ID 2. NAME 3. CLASS NAME 4.LINKED TEXT 5. XPATH 6.CSS**

**7. PARTIAL LINKED TEXT**

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

****

* **It does not have built-in Object Repository like UTF/QTP to maintain objects/elements in centralized location. However, we can overcome this limitation using Page Object Model.**

**\*\*\*Locators in Selenium:-**

-**Locators** provide a way to access the HTML elements from a web page.

-In Selenium, we can use locators to perform actions on the text boxes, links, check-boxes and other web elements.

-They are the basic building blocks of a web page.

- **The choice of locator depends largely on your Application Under Test.**

**-** When we don't have an option to choose Id or Name, we should prefer using CSS locators as the best alternative.

- **Locator** is a command that tells Selenium IDE which GUI elements ( say Text Box, Buttons, Check Boxes etc) its needs to operate on.

* CSS has more Advantage than Xpath
* CSS is much more faster and simpler than the Xpath.
* In IE Xpath works very slow, where as Css works faster when compared to Xpath.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***Types of LOCATORS\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

1. **ID**🡪By.id(“Value”)
2. **Name**🡪By. name(“Value”)
3. **ClassName** 🡪 By.className (“Value”)
4. **LinkText**🡪By. linkText (“Displayed Text”)🡪 LinkText shows hyper-link and its tag-name is<a>.
5. **CSS**🡪Cascading Style Sheet

Syntax->TagName[Property=’PropertyValue’]

For Example : input[type=’Submit’]

**jQuery UI** is a mixture of methods and a set of user interface effects, widgets, interactions and themes which can be provided in the web page using jQuery methods. If you want to build up a powerful web application that includes various features such as dragging, dropping, date picker, tooltips, etc. then jQuery UI is a perfect choice to build up these effects.  
In this article, we are going to learn about various jQuery UI interactions.

**Draggable() Method**

This method allows the elements to be dragged with the help of mouse. Using jQuery UI, we can make the DOM(**D**ocument **O**bject **M**odel) elements to drag anywhere within the view port. This can be done by clicking on the draggable object by mouse and dragging it anywhere within the view port.  
**Syntax:**  
The draggable() method has two forms and the use of each form depends on the requirement. These are as follows :-

$(selector, context).draggable (options);

$(selector, context).draggable ("action", [params]);

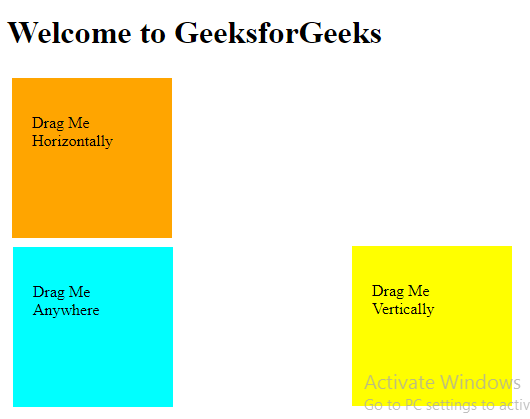
**The following table shows the different options that can be used with this method:**

| OPTION | PURPOSE |
| --- | --- |
| addClass | If the value of this option is set to false, it will prevent the DOM elements to be dragged . The default value this option is true. |
| axis | This option is used constrain the movement of the draggable object. If the value of this option is set to Y , then the object can be dragged in the vertical direction only and if the value of this option is set to X , then the object can be dragged into horizontal direction only. |
| containment | This option is also used constrain the movement of the draggable object within the specific region or some element.The default value this option is false. |
| opacity | This option is used to control the opacity of the draggable object while it is dragged.The default value this option is false. |

**Example:**  
In this example, the <div> with id=”d1″ can be dragged anywhere within the view port, <div> with id=”d2″ can be dragged along X axis and <div> with id=”d3″ can be dragged along Y axis.  
**Code #1:**

|  |
| --- |
| <!doctype html>  <html>  <head>  <title>jQuery UI Draggable</title>  <link rel="stylesheet" href="//code.jquery.com/ui/                              1.12.1/themes/base/jquery-ui.css">  <link rel="stylesheet" href="/resources/demos/style.css">  <style type="text/css">  #d1 {      width: 120px;      height: 120px;      background-color :aqua;      padding:20px;      float:left;      margin:5px;      }  #d2 {      width: 120px;      height: 120px;      background-color :orange;      padding:20px;      float:left;      margin:5px;      }  #d3 {      width: 120px;      height: 120px;      background-color :yellow;      padding:20px;      float:left;      margin:5px;      }  </style>  <script src="<https://code.jquery.com/jquery-1.12.4.js>"></script>  <script src="<https://code.jquery.com/ui/1.12.1/jquery-ui.js>">  </script>  </head>  <body>  <h1>Welcome to GeeksforGeeks</h1>  <div id="d1"><p>Drag Me Anywhere</p></div>  <div id="d2"><p>Drag Me Horizontally</p></div>  <div id="d3"><p>Drag Me Vertically</p></div>  <script type="text/javascript">  $( function() {      $("#d1").draggable();  } );  $( function() {      $("#d2").draggable({axis:"x"});  } );  $( function() {      $("#d3").draggable({axis :"y"});  } );  </script>  </body>  </html> |

**Output:**  
Before Dragging  


After Dragging  


**Droppable() Method:**

This method allows the elements to be dropped with the help of mouse. Using jQuery UI, we can make the DOM(**D**ocument **O**bject **M**odel) elements to drop anywhere within the view port on the specified target. This can be done by clicking on the draggable object by mouse and drop it on the specified target.  
**Syntax:**  
The droppable() method has two forms and the use of each form depends on the requirement. These are as follows :-

$(selector, context).droppable (options)

$(selector, context).droppable ("action", params)

**The following table shows the different options that can be used with this method:**

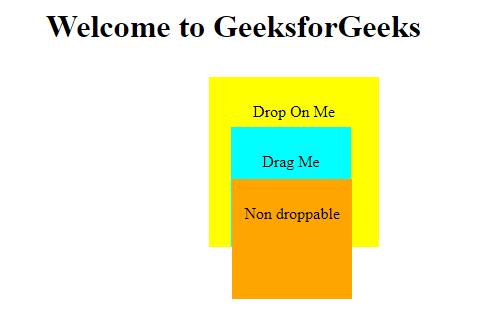
| OPTION | PURPOSE |
| --- | --- |
| accept | The value of this option specifies that which draggable objects can be dropped on the specified target. The default value of this option is \*. |
| addClass | If the value of this option is set to false, it will prevent the DOM elements to be dropped . The default value this option is true. |
| disable | This option is also used to disable the dropppable property of the DOM element.If the value of this option is set to ture , then the object cannot be dropped and if the value of this option is set to false, then the object can be dropped on the specified target. |

**Example :**  
In this example, the <div> with id=”drag” is dragged and dropped over the <div> with id=”drop”.  
**Code #1:**

|  |
| --- |
| <!doctype html>  <html lang="en">  <head>  <title>jQuery UI Droppable</title>  <link rel="stylesheet" href="//code.jquery.com/ui/1.12.1/                               themes/base/jquery-ui.css">  <link rel="stylesheet" href="/resources/demos/style.css">  <style type="text/css">  #drag      {      width: 100px;      height: 100px;      float: left;      margin: 10px;      background-color :aqua;      padding:10px;      }  #drop      {      width: 150px;      height: 150px;      float: left;      margin: 10px;      background-color:yellow;      padding:10px;      }  </style>  <script src="<https://code.jquery.com/jquery-1.12.4.js>"></script>  <script src="<https://code.jquery.com/ui/1.12.1/jquery-ui.js>">  </script>  <script>  $( function() {      $( "#drag" ).draggable();      $( "#drop" ).droppable(          {              drop :function()          {              alert("I am dropped");          }          } );          } );  </script>  </head>  <body>  <center>  <h1 align="center">Welcome to GeeksforGeeks</h1>  <div id="drag">  <p>Drag Me</p>  </div>  <div id="drop">  <p>Drop On Me</p>  </div>  </center>  </body>  </html> |

**Output:**  
Before Dropping  
  
After Dropping  
  
**Code #2:**  
In this example, the <div> with id=”drag” is dragged and dropped over the <div> with id=”drop” and it cannot be dropped over the <div> with id=”non-drop”.

|  |
| --- |
| <!doctype html>  <html lang="en">  <head>  <title>jQuery UI Droppable</title>  <link rel="stylesheet" href="//code.jquery.com/ui/1.12.1/                               themes/base/jquery-ui.css">  <link rel="stylesheet" href="/resources/demos/style.css">  <style type="text/css">  #drag      {      width: 100px;      height: 100px;      float: left;      margin: 10px;      background-color :aqua;      padding:10px;      }  #non-drop          {      width: 100px;      height: 100px;      float: left;      margin: 10px;      background-color :orange;      padding:10px;      }  #drop      {      width: 150px;      height: 150px;      float: left;      margin: 10px;      background-color:yellow;      padding:10px;      }    </style>  <script src="<https://code.jquery.com/jquery-1.12.4.js>"></script>  <script src="<https://code.jquery.com/ui/1.12.1/jquery-ui.js>">  </script>  <script>  $( function() {      $( "#drag" ).draggable();      $( "#non-drop" ).draggable();          $( "#drop" ).droppable(          {              accept:"#drag",              drop :function()          {              alert("I am dropped");          }          } );          } );    </script>  </head>  <body>  <center>  <h1 align="center">Welcome to GeeksforGeeks</h1>  <div id="drag">  <p>Drag Me</p>  </div>  <div id="non-drop">  <p>Non droppable</p>  </div>  <div id="drop">  <p>Drop On Me</p>  </div>  </center>  </body>  </html> |

**Output:**  
Before Dropping  
  
After Dropping  


\*Alert:-An Alert in Selenium is a small message box which appears on screen to give the user some information or notification.

\*\*\*It notifies the user with some specific information or error, asks for permission to perform certain tasks and it also provide warning messages as well.

\*\*\*\*\*\*\*\*\*To Display Data:-

driver().switchTo().alert().getText();

\*\*\*\*\*\*\*\*\*\*Click on Ok button:-

driver().switchTo().alert().accept();

\*\*\*\*\*\*\*\*\*\*Click on Cancel button:-

driver().switchTo().alert().dismiss();

\*\*\*\*\*\*\*\*Send data in Alert:-

driver().switchTo().alert().sendkeys(”OTP”);