Selenium IDE

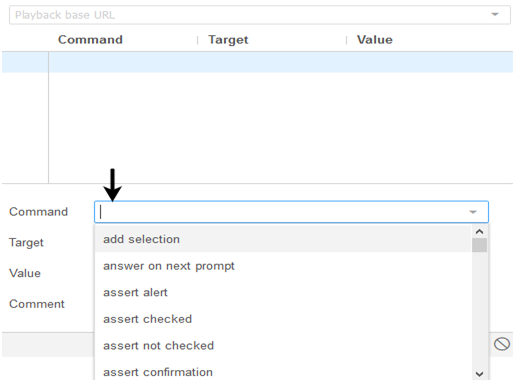
In this section, you will learn how to create a test case manually using Selenium commands in Selenium IDE. In simple words, we will create test cases by inserting selenium commands instead of recording option.

For this test, we will search a text operation on any publically available search engine (say "Google"). Subsequently, we will create a Login test case in the same test suite.

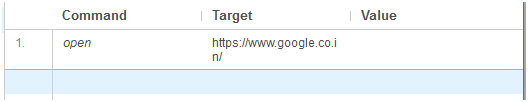
To create a test case manually, first you have to go through the most commonly used selenium commands which we have discussed in the previous section. Now, we will create our first test case on search operation. We will search our text on Google search engine.

1. Insert Commands

* Launch Firefox browser.
* Click on the Selenium icon present on the top right corner on your browser.
* It will launch the default interface of Selenium IDE.
* Enter the project name as "Manual Test".
* Enter the test case name as "Search Test".
* Click on the command text box present on the Test Script Editor Box.



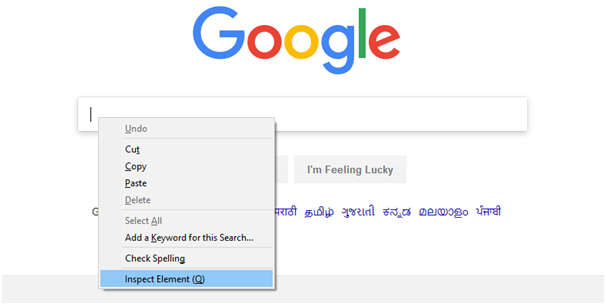
* Modify the properties of First command as:
* Command : open
* Target : [https://www.google.co.in](https://www.google.co.in/)
* During execution of the test case, this command will load the Google search engine web page on your Firefox browser.



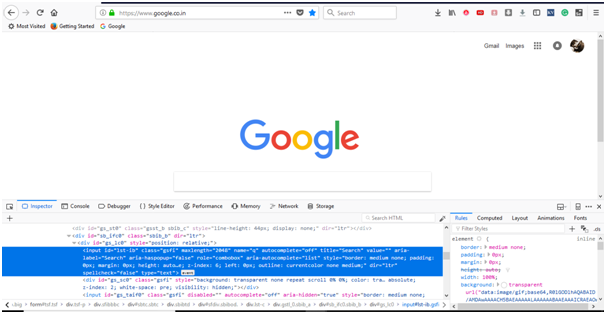
Now, we have to add a command that will click on the Google search engine text box. For this, we need a unique identification element of the text box which will help the IDE to identifythe target location.

The method for finding a unique identification element involves inspection of HTML codes.

* Open URL:[https://www.google.co.in](https://www.google.co.in/)your Firefox browser.
* Right click on the Google search text box and select Inspect Element.



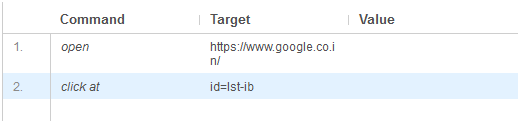
* It will launch a window containing all the specific codes involved in the development of the test box.



* Pick the input tag element that contains an ID name for the text box.

IDE-Creating Test Cases Manually

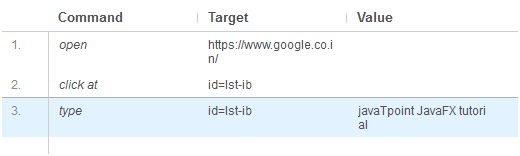
* Modify the properties of second command as:
* Command :click at
* Target : id=lst-ib
* During execution of the test case, this command will click on the search text box present on the Google search engine web page.



Note: The "Value" portion of the Test Script Editor box is optional for most of the commands.

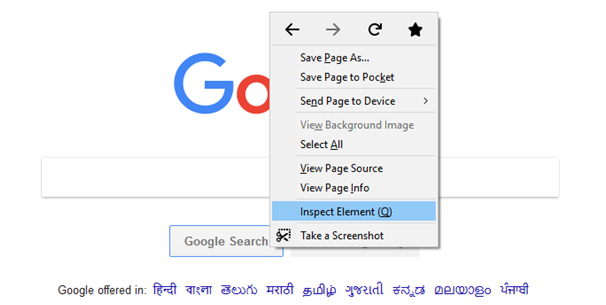
We will use the same identification ID for our third command. The third command will type the specified text in to the Google search text box.

* Modify the properties of third command as:
* Command : type
* Target : id=lst-ib
* Value: javaTpointJavaFX tutorial
* During execution of the test case, this command will type the specified text on the Google search text box.

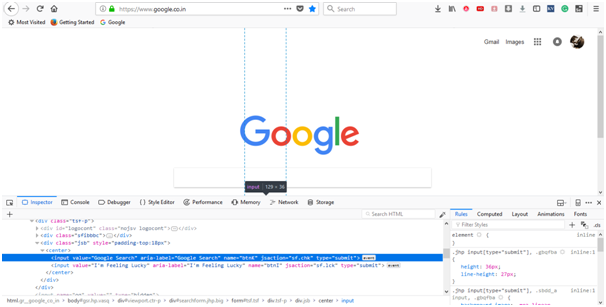


We will now add a command which will generate a button click event on our web page. For this event to be generated, we need a unique identification element for the Google search button.

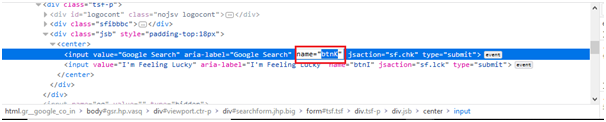
* Right click on the Google search button and select Inspect Element.



* It will launch a window containing all the specific codes involved in the development of the search button.



* Pick the name element that contains the specified name for the Google search button.



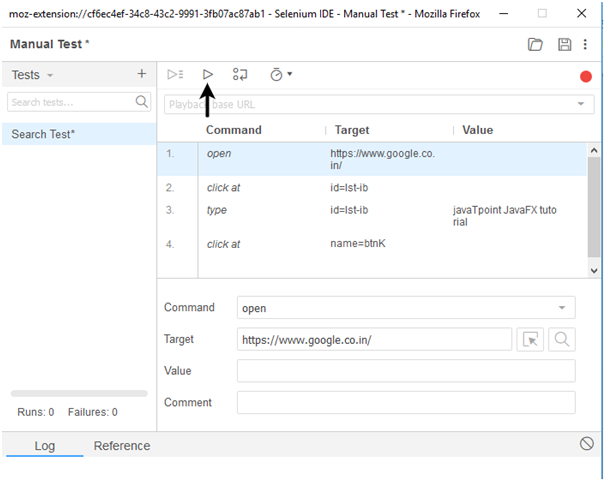
* Modify the properties of fourth command as:
* Command : click at
* Target : name=btnK
* During execution of the test case, this command will click on the search button present on the Google search engine web page.



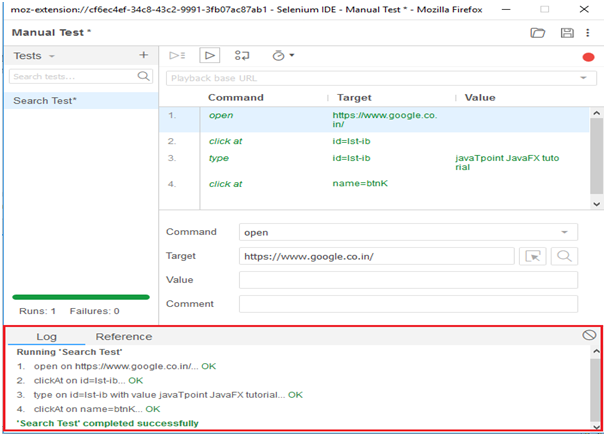
We are now ready to execute our first test script.

Executing the Test Script

* Click on the "Run Current Test" button present on the tool bar menu of the IDE. It will execute all of your inserted commands on the browser and gives you an overall summary of the executed test script.



* The Log pane displays the overall summary of the executed test scripts.



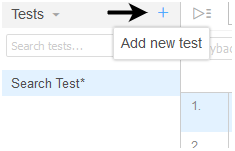
Now, we will create our second test case within the same test suite.

We will generate a test case based on login feature provided by one of the most popular website "Rediffmail".

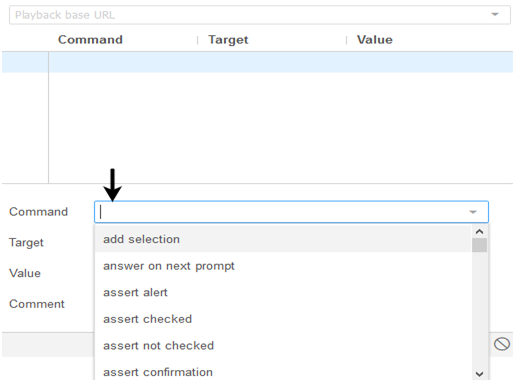
First you have sign-up to get the login credentials. For this test, we have already generated our login credentials.

1. Insert Commands

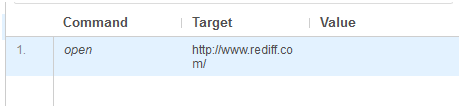
Click on the "Add new test" button at the top of the test case pane.



* Rename the test case as "Login Test".
* Click on the command text box present on the Test Script Editor Box.



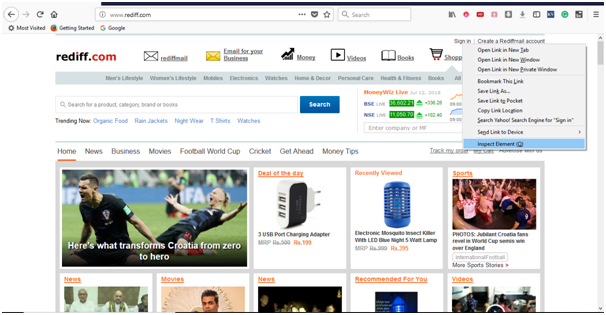
* Modify the properties of First command as:
* Command : open
* Target :[http://www.rediff.com/](http://www.rediff.com/" \t "_blank)
* During execution of the test case, this command will load the Rediff home page on your Firefox browser.



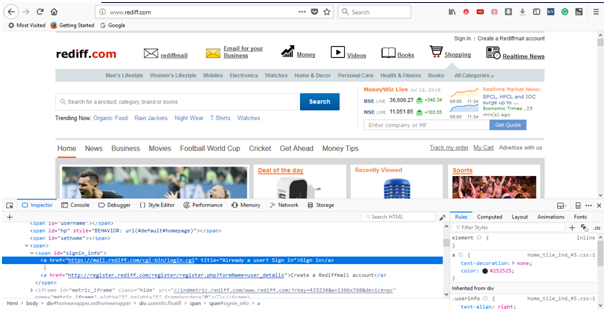
Now, we have to add a command that will click on the "Sign-in" link present at the top right corner of Rediff website. For this, we need a unique identification element of the "Sign-in" link which will help the IDE to identify the target location.

The method for finding a unique identification element involves inspection of HTML codes.

* + Open URL:<http://www.rediff.com/> on your Firefox browser.
  + Right click on the "Sign-in" and select Inspect Element.



* + It will launch a window containing all the specific codes involved in the development of the "Sign-in" link.



* + Pick the link element that contains the specified name for the "Sign-in" link.

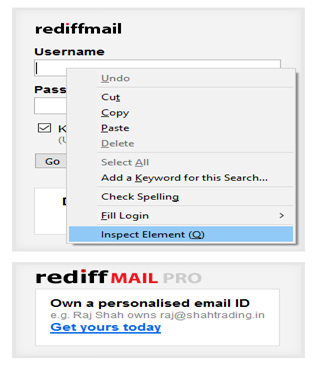


* + Modify the properties of second command as:
  + Command : click at
  + Target : link=Sign in
  + During execution of the test case, this command will click on the "Sign in" link.

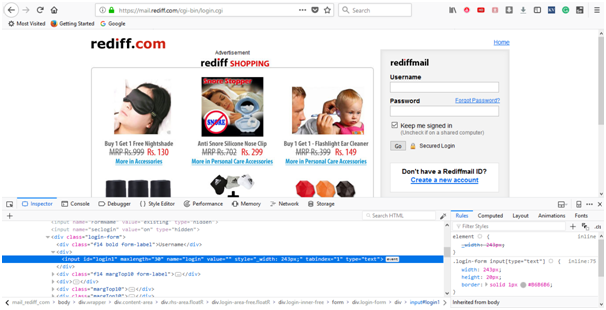


The "Sign in" link will redirect you to the login page. Therefore, for the third command to be entered we need a unique identification for the "Username" text box which will help the IDE to identify the target location.

* + Right click on the "Username" text box and select Inspect Element.



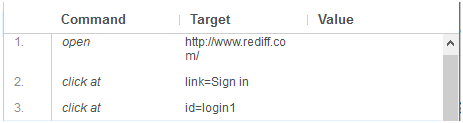
* + It will launch a window containing all the specific codes involved in the development of the "Username" text box.



* + Pick the ID element that contains the specific ID for the "Username" text box.



* + Modify the properties of third command as:
  + Command : click at
  + Target : id=login1
  + During execution of the test case, this command will click on the "Username" text box.



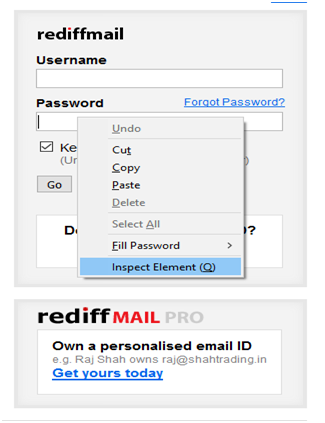
We will use the same ID element for our next command which involves typing the user-id as login credential.

* + Modify the properties of fourth command as:
  + Command : type.
  + Target : id=login1
  + Value : frea\*\*\*\*\*\*\*\*\* (User Login ID)
  + During execution of the test case, this command will type the user id in the "Usename" text box.

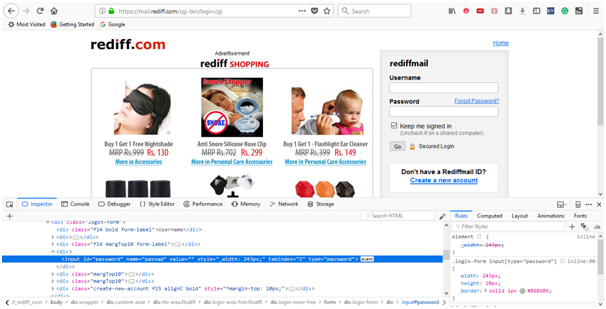


We will now add a command that will click on the "Password" text box. For this command, we need a unique identification for the "Password" field which will help the IDE to identify the target location.

* + Right click on the "Password" text box and select Inspect Element.



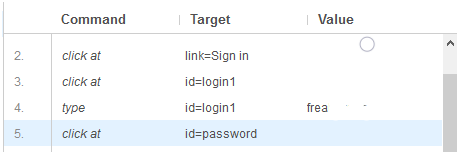
* + It will launch a window containing all the specific codes involved in the development of the "Password" text box.



* + Pick the ID element that contains the specific ID for the "Password" text box.

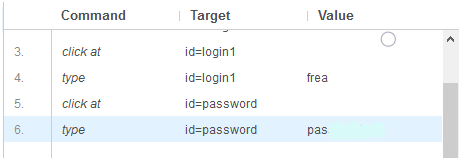


* + Modify the properties of fifth command as:
  + Command : click at
  + Target : id=password
  + During execution of the test case, this command will click on the "Username" text box.



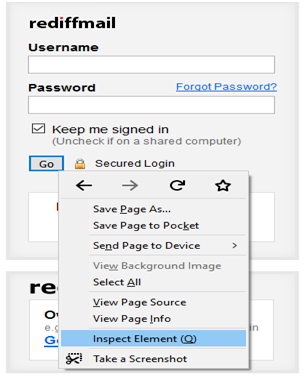
We will use the same ID element for our next command which involves typing the password as login credential.

* + Modify the properties of sixth command as:
  + Command : type
  + arget : id=password
  + Value : \*\*\*\*\*\*\*\*\*\*(User Login Password)
  + During execution of the test case, this command will type the user login password in the "Password" text box.

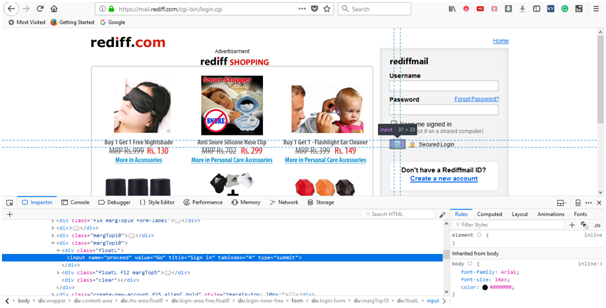


At last, we need a unique identification element for the Login Submit button which will help the IDE to identify the target location.

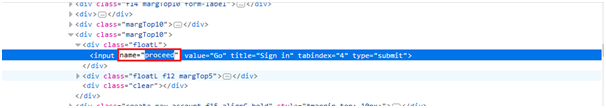
* + Right click on the "Go" button and select Inspect Element.



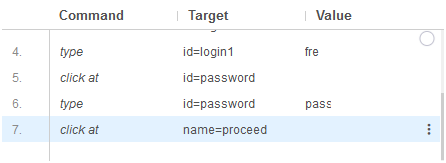
* + It will launch a window containing all the specific codes involved in the development of the "Go" submit button.



* + Pick the name element that contains the specific name for the "Go" submit button.



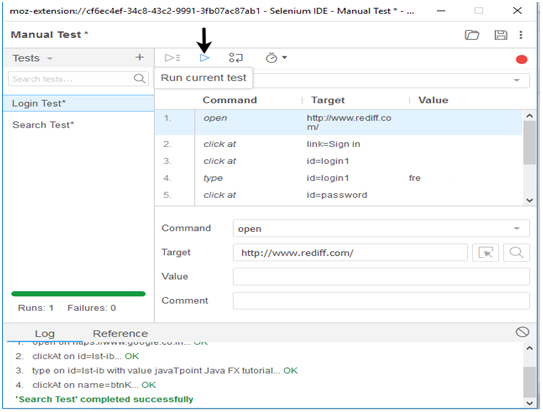
* + Modify the properties of seventh command as:
  + Command : click at
  + Target : name=proceed
  + During execution of the test case, this command will click on the "Go" submit button.



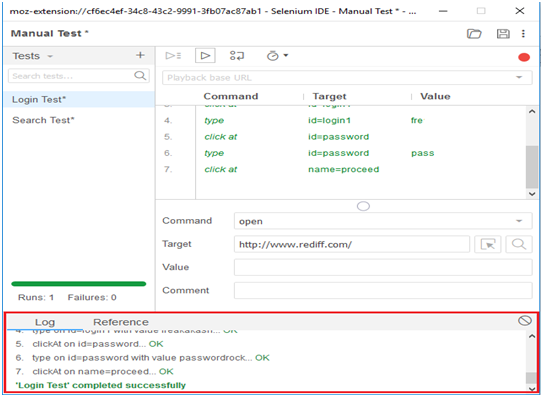
We are now ready to execute our second test script.

Executing the Test Script

* + Click on the "Run Current Test" button present on the tool bar menu of the IDE. It will execute all of your inserted commands on the browser and gives you an overall summary of the executed test script.

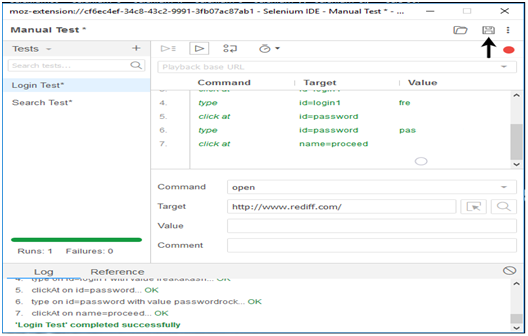


* + The Log pane displays the overall summary of the executed test scripts.

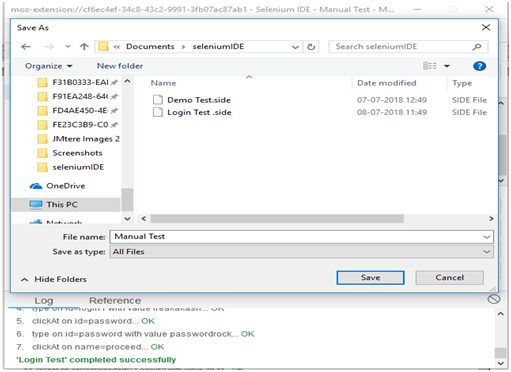


Saving the test suite

* + Click on the save button present on the extreme right corner of the menu bar.



* + Save the entire test suite as "Manual Test".



* + The test suite can be found at the location provided in the above steps.