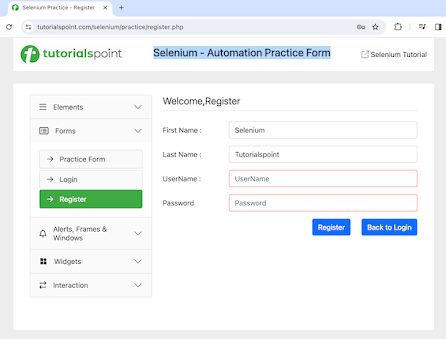
**Lab Exercise 6 - Selenium IDE - Store Variables**

Selenium Integrated Development Environment can be used to store and access variables while creating a test for automation. To store variables and to access them, we can use the **store** and its related commands in Selenium IDE. We use the command **echo** to print the stored value of the variable under the Logs tab in Selenium IDE. We can also access an element on a web page and obtain its text using the **store Text** command and obtain the value of an element using the **store Value** command.

Example

Let us take an example of the below page, where we would enter the text **Selenium** beside the **First Name:** label and **Tutorialspoint** beside **Last Name:** label. Also, we would obtain the text **Selenium - Automation Practice Form** using the Selenium IDE taking the help of the store variables feature of this tool.



Storing and Accessing Variables in Selenium IDE

The steps to store variables and access them in Selenium IDE are listed below −

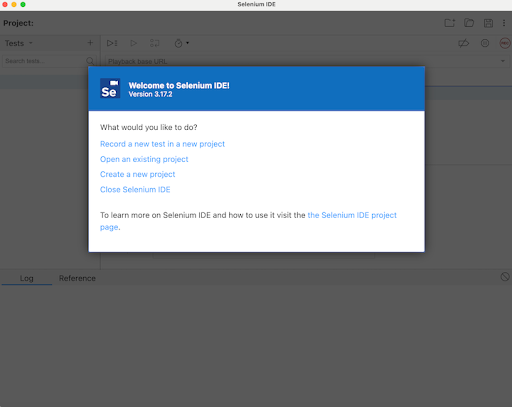
**Step 1** − Click on the Selenium IDE extension visible on the browser after its installation.

**Step 2** − Selenium IDE should be launched along with it, a welcome message should be displayed with the version of the tool. In the below image, the version installed is 3.17.2.

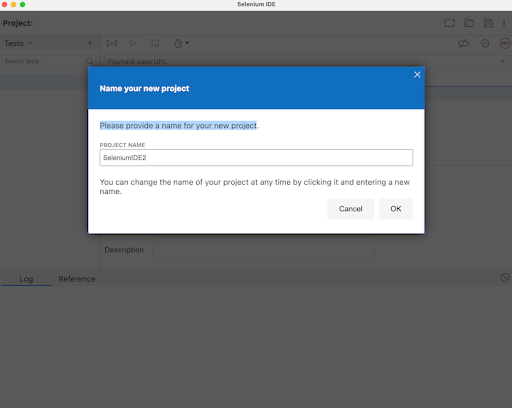
Along with this, it gives us the option to select what we would like to do with the tool, like Record a new test in a new project, Open an existing project, Create a new project, and Close Selenium IDE.

Also, a link with the text - **the Selenium IDE project page** is provided, which on clicking would land us to the Selenium Integrated Development Environment documentation page.

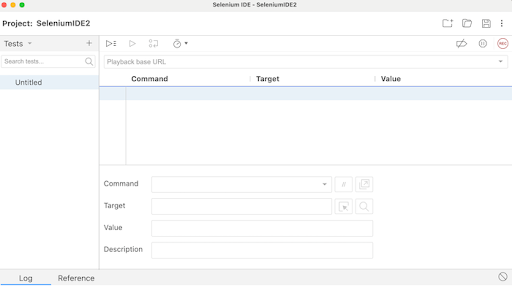
<https://www.selenium.dev/selenium-ide/>



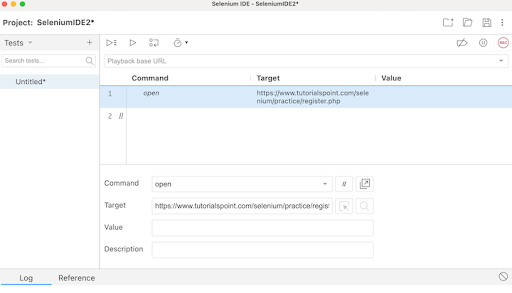
**Step 3** − Click on the Create a new project link, then enter a project name under the **Please provide a name for your new project**. Finally, we would click OK.



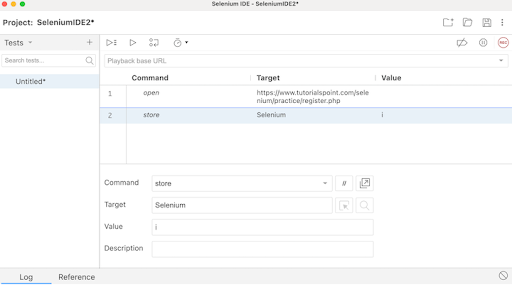
**Step 4** − We would have the workspace ready in the Selenium IDE as shown in the below image.



**Step 5** − Enter open in the Command field, and [https://www.tutorialspoint.com/selenium/](https://www.tutorialspoint.com/selenium/practice/register.php) in the Target field, to launch the application.



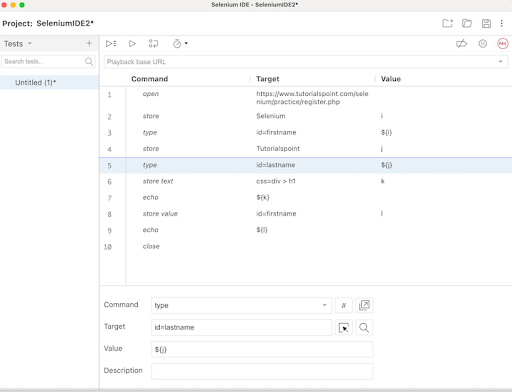
**Step 6** − Enter store in the Command field, Selenium in the Target field, and i in the Value field. Please note, Selenium is the value we would like to enter using the variable i.



**Step 7** − Enter type in the Command field, id=firstname in the Target field, and ${i} in the Value field. Please note, id=firstname is the locator value of the input box and ${i} is used to refer to the variable value we would enter using the variable i.

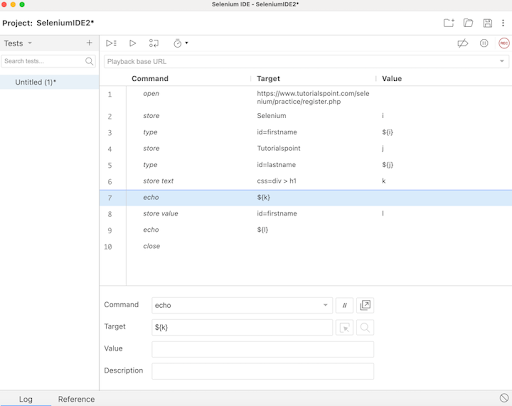
**Step 8** − Enter store in the Command field, Tutorialspoint in the Target field, and j in the Value field. Please note, Tutorialspoint is a value we would like to enter using the variable j.

**Step 9** − Enter type in the Command field, id=lastname in the Target field, and ${j} in the Value field. Please note, id=lastname is the locator value of the input box and ${j} is used to refer to the variable value we would enter using the variable j.



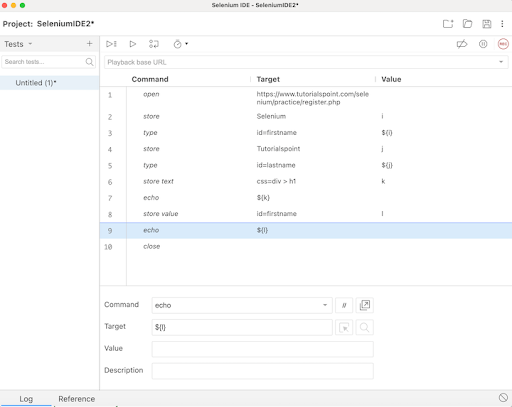
**Step 10** − Enter store Text in the Command field, css=div > h1 in the Target field, and k in the Value field. Please note, css=div > h1 is the locator value of the text that we would obtain and k is used to refer to the variable value where we would store the text obtained.

**Step 11** − Enter echo in the Command field, ${k} in the Target field. Please note, this would print the text we accessed in Step10 on the web page under the Logs in Selenium IDE.

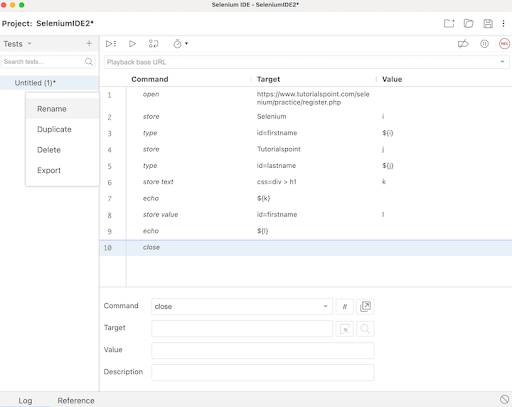


**Step 12** − Enter store value in the Command field, id=firstname in the Target field, and l in the Value field. Please note, id=firstname is the locator value of the input box and l is used to refer to the variable we would use to obtain the value entered in the input box.

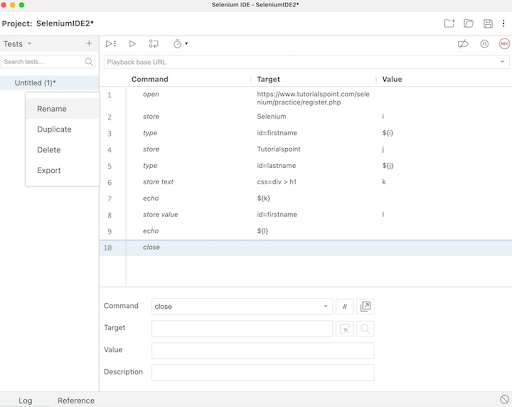
**Step 13** − Enter echo in the Command field, ${l} in the Target field. Please note, this would print the value entered in Step7 on the web page under the Logs in Selenium IDE.



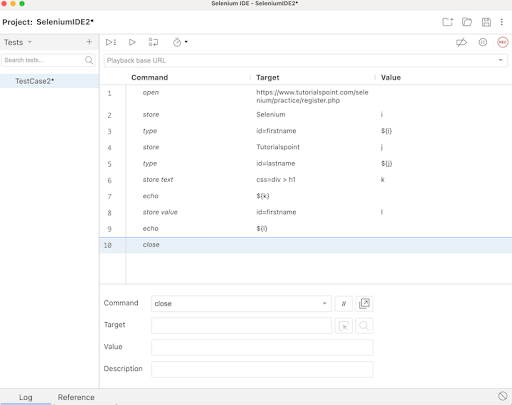
**Step 14** − Enter close in the Command field to close the browser window.



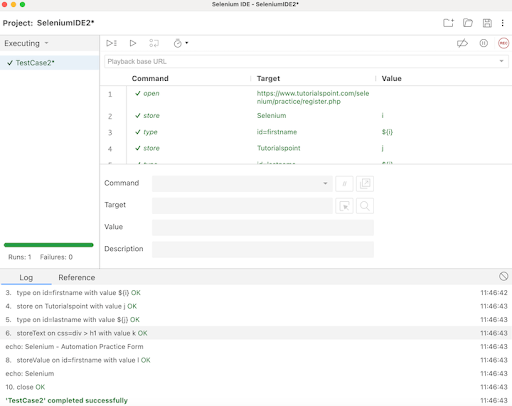
**Step 15** − Click on the three dots appearing at the left pane, then select the Rename option.



**Step 16** − Enter a name, say TestCase2 under the Rename Test case field, then click on Rename. The entered name would appear on the left of Selenium IDE.



**Step 17** − Click on the Run all tests from the top, and wait for the test execution to complete. In our example, we would see Runs: 1, Failures: 0, and a green bar denoting the test ran successfully without any failures. Also, the message TestCase2 completed successfully appeared under the Logs.

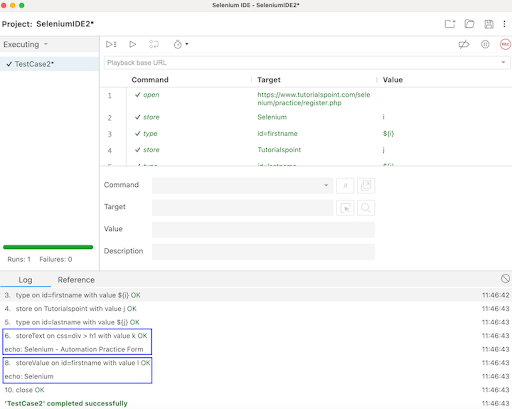


In the example above, we obtained the text **Selenium - Automation Practice Form** which appeared on the web page, and also retrieved the value entered in the input box beside the **First Name:** label - **Selenium** under the Logs. Besides, we got a green tick beside the test case name - **TestCase2** at the top, signifying a passed test.

Apart from that, all the test steps, along with the test case name turned green, reflecting a PASS for all the steps. All the steps which reflected under the Log tab, showed an OK in green.

Thus, in Selenium IDE, we would get the information about the test results both at the test step level and only if all the steps would be executed successfully, we would get a PASS at the test case level as well.

Moreover, we would infer that variables used in one step can be accessed further in other steps using the syntax: ${<variable name>}. In the Steps6, 7, 8, 9, 10, 11, 12, and 13, we followed this approach in our test. Also, we had used store and store related commands like store Text, store Value, and so on to store the variables. The echo command was used to print those values under the Logs tab.



Thus, in this tutorial, we had discussed how to store variables using the Selenium IDE.