PRACTICAL NO . 1

Aim : Install Selenium IDE; Write a test suite containing minimum 4 test cases for different formats.

Installation of Selenium IDE in Chrome Browser

Step-1 : Launch Chromium browser.

Step-2 : Search for Selenium IDE in the ”chrome web store”.

Step-3 : Select ”Add to Chrome” Button which will lead to chromium pop-up asking for permission to add the extension, click ”Add Extension” button.

Step-4 : Chromium will automatically download and install the extension. Which can be accessed from the ”Extensions” Icon Button on the Navigation Bar.

Step-5 : Clicking on the selenium IDE icon, the extension will open a new window with the
selenium IDE.

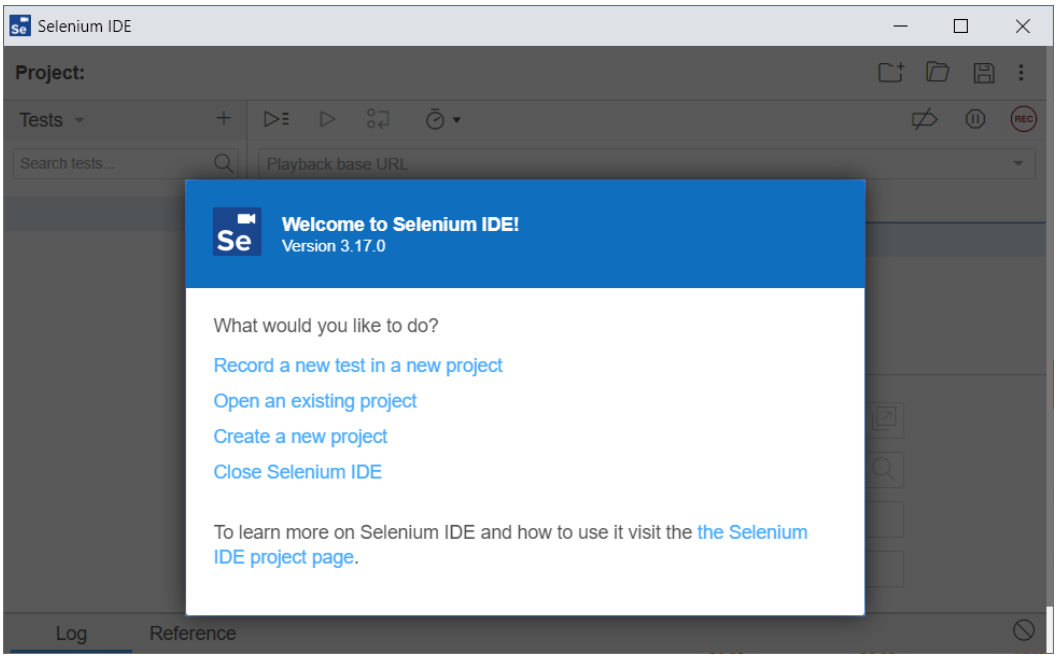


Figure 1 : Selenium IDE Window

Selenium IDE :

Selenium Record and Playback tool for ease of getting acquainted with Selenium WebDriver.

The new Selenium IDE is designed to record your interactions with websites to help you generate and maintain site automation, tests, and remove the need to manually step through repetitive takes.

Features include:

1.Recording and playing back tests on Firefox and Chrome.

2.Organizing tests into suites for easy management.

3.Saving and loading scripts, for later playback. Support for Selenium 3.

Detailed Selenium IDE Features :

1 . Menu Bar Menu bar is positioned at the top most portion of the Selenium IDE interface. It allows the user to change name/open project/save project and more.

Figure 2: Menu Bar

2 . ToolBar The toolbar offers options such as play test/record test and play all the tests in test suite. More advanced features such as breakpoints/debugger are also available.

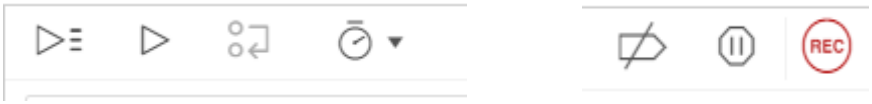


Figure 3: Tool bar

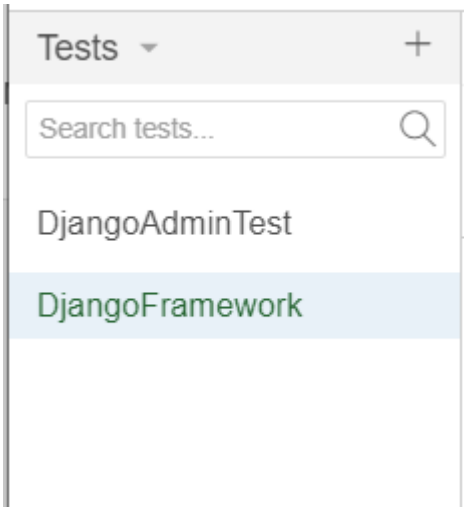
3 . Test Case panel This panel allows the user to add/delete /modify the selenium commands. The Command, Target, and Value entry fields display the currently selected command along with its parameters. These are entry fields where you can modify the currently selected command. The first parameter specified for a command in the Reference tab of the bottom pane always goes in the Target field. If a second parameter is specified by the Reference tab, it always goes in the Value field. The Comment field allows you to specify in a human readable format a description of the current command.

	Command	Target	Value
1	*open*	http://localhost:8000/	
2	*set window size*	1552x840	
3	*click*	css=.one h4	
4	*store window handle*	root	
5	*select window*	handle=${win3269}	
6	*close*		
7	*select window*	handle=${root}	
8	*click*	css=.two h4	
9	*select window*	handle=${win2011}	
10	*select window*	handle=${root}	

Figure 4: Test Case Panel

Figure 5: Script Command Editor

4 . Navigation Panel Navigation between test cases and test suites is done through the right hand side of Selenium IDE. Clicking on Tests with the small caret will open up a menu. When saving the project will be saved as the new .side format, which will include all test cases and suites combined. It is refered to as a project.



5 . Console Panel The bottom pane, called Console Panel for it's similarity with web dev tools, is used for different utility functions: Log, Reference, depending on which tab is selected.

5.1 Log When you run your test case, error messages and information messages showing the progress are displayed in this pane automatically, even if you do not first select the Log tab. These messages are often useful for test case debugging. Notice the Clear button for clearing the Log.

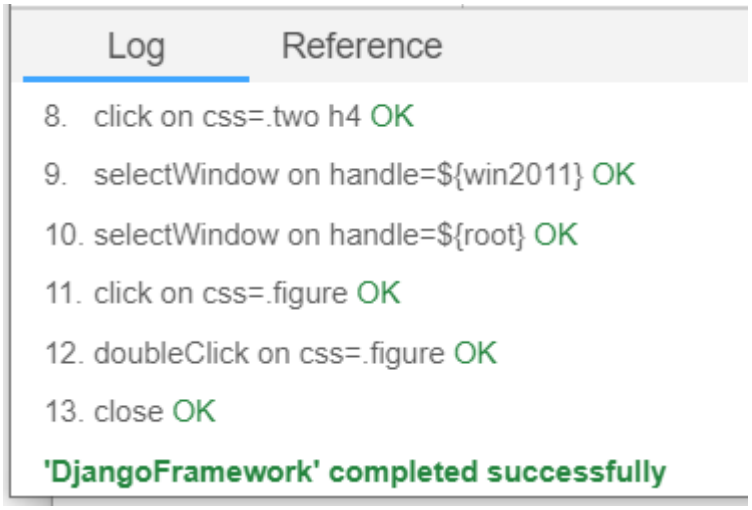


Figure 7: Log Tab

5.2 Reference When adding, editing, or viewing test steps this pane displays information about the command that's currently in focus. Details like name, description, what arguments it accepts, and details about those arguments.

Figure 8: Reference Tab

Creating/Recording and Running a Test Suite :

Step-1 :- Launch the Chromium browser and open the selenium IDE extension.

Step-2 :- When the Welcome screen of selenium IDE is visible, select the option ”create a new project”.

Step-3 :- Give an appropriate name to the project and rename the untitled default test case to another appropriate title.

Step-4 :- Set the ”Playback base URL” to the URL value of the website/webpage you wish to test.

Step-5 :- At this point you have two choices where in you can either manually type the test steps in the IDE or let selenium record the steps for you by clicking the ”Record button ON”.

Step-6 :- We will first Record the steps using selenium’s Record functionality. On clicking the ”Record button ON” or keyboard shortcut ”Ctrl-U”.

Step-7 :- A new window will pop up with the base URL provided by us earlier , this window is special because selenium will record our mouse clicks and key inputs. Be careful not to reveal any sensitive information in the record mode.

Step-8 :- Now perform the testing on various elements of the HTML document either by direct clicks/key inputs or by selecting various methods from the ”right-click → selenium → drop-down menu” which will provide various options for assert various elements and validate various values.

Step-9 :- You can try out filling forms, dummy login credential test and a lot more.

Step-10 :- After you are done with the test script click on ”Stop Recording” button or keyboard shortcut ”Ctrl-U”.

Step-11 :- Save the test script by clicking on ”Save project icon” or keyboard shortcut ”Ctrl-S”.

Step-12 :- Next save the script in the desired folder path and selenium will download the ”.side” file to the same .

Step-13 :- You can reopen the saved Project by clicking ”Open Project icon” or keyboard Shortcut ”Ctrl-O” and selecting the ”.side” file from the desired location.

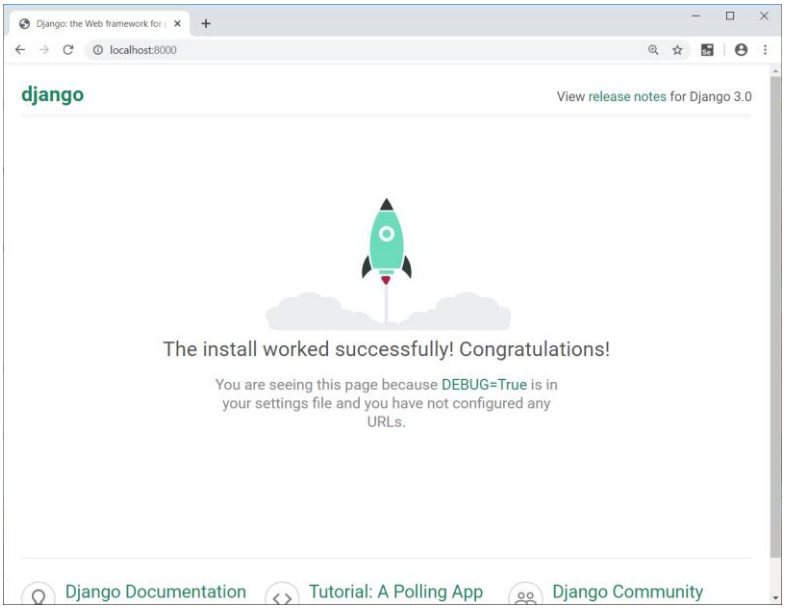
Step-14 :- To run the test click on ”Playback icon” button and selenium will run the test as per the script and log all the events in the window. You can verify which steps succeed and which ones fail with Error messages as well.

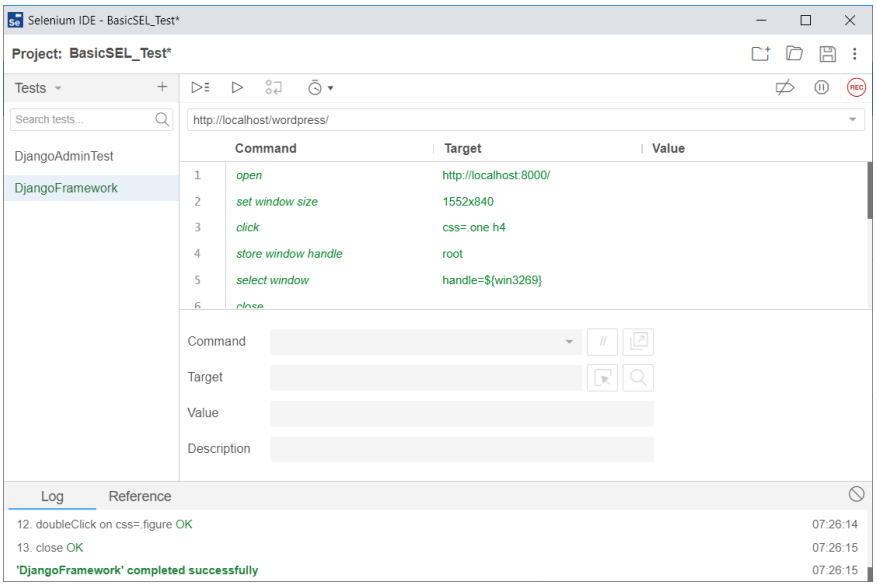
Step-15 :- The test script can be written manually as well and will produce the same results.

Test Case 1 :

Test URL : http://localhost:8000/

Command 1 -> Set Window Size.
Command 2 -> Store window handle.
Command 3 -> Select window.
Command 4 -> Double click.

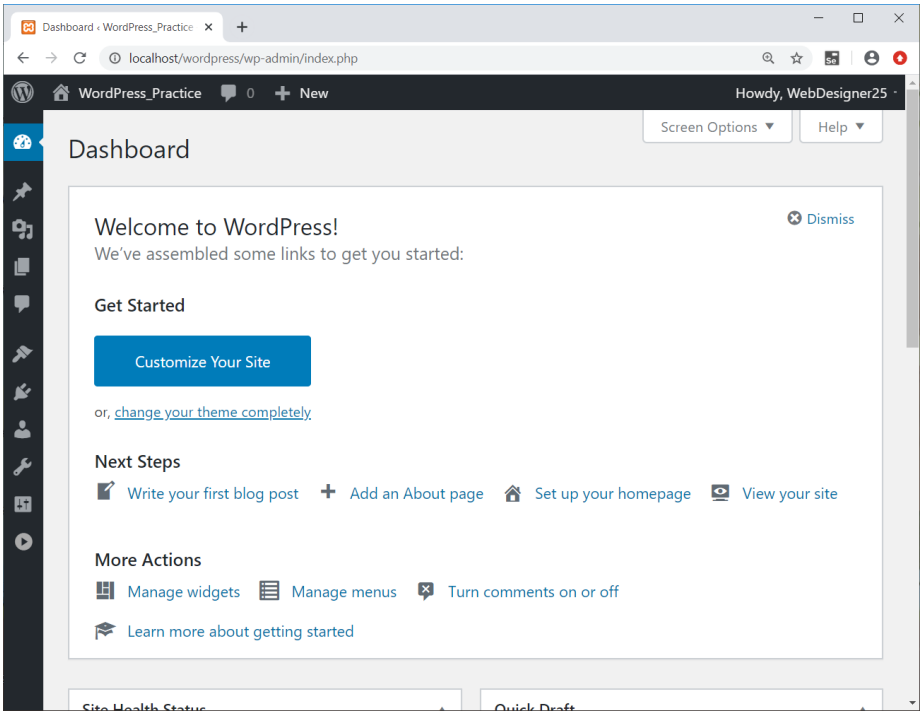


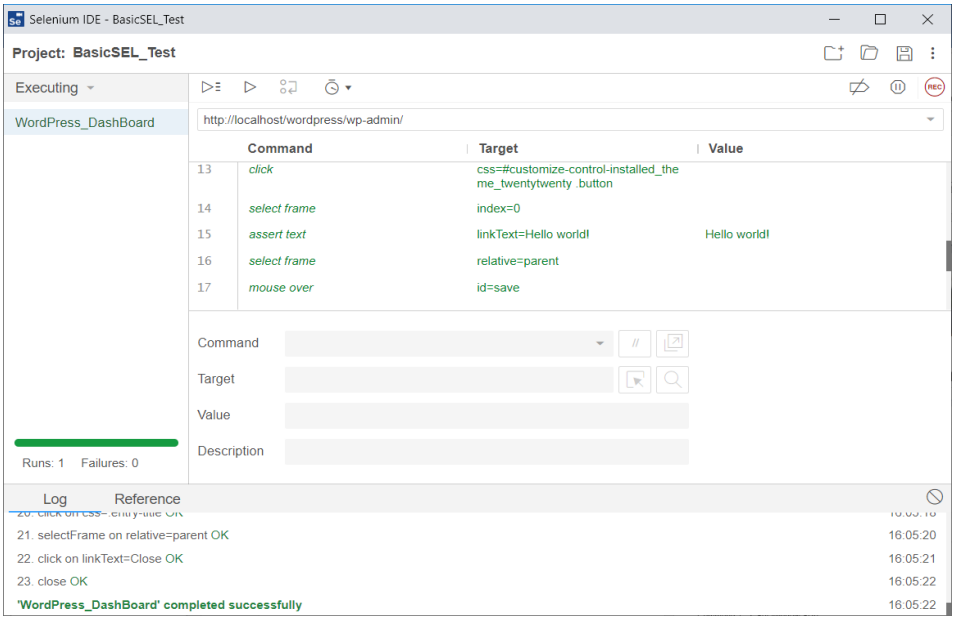


Test Case 2 :

Test URL : http://localhost/wordpress/wp-admin/index.php/

Command 1 -> Set Window Size.
Command 2 -> Set frame
Command 3 -> Assert text.
Command 4 -> Select frame.
Command 5 -> Mouse Over.

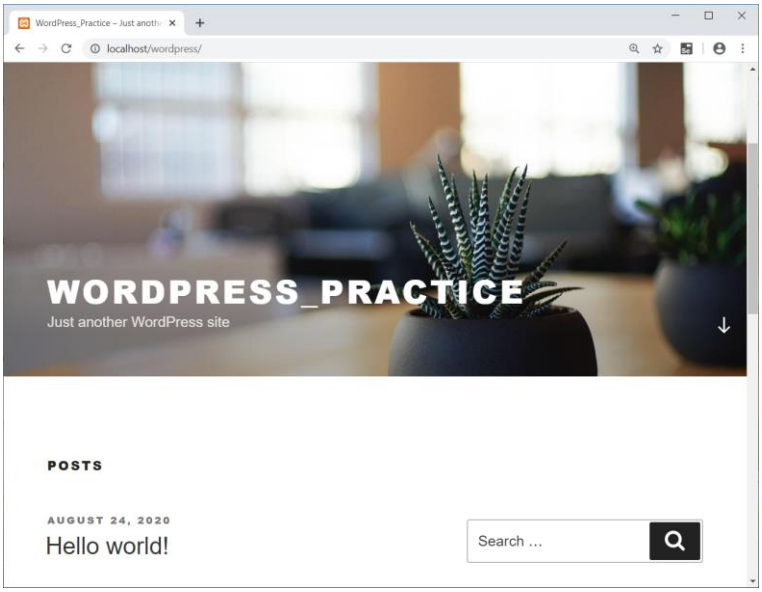


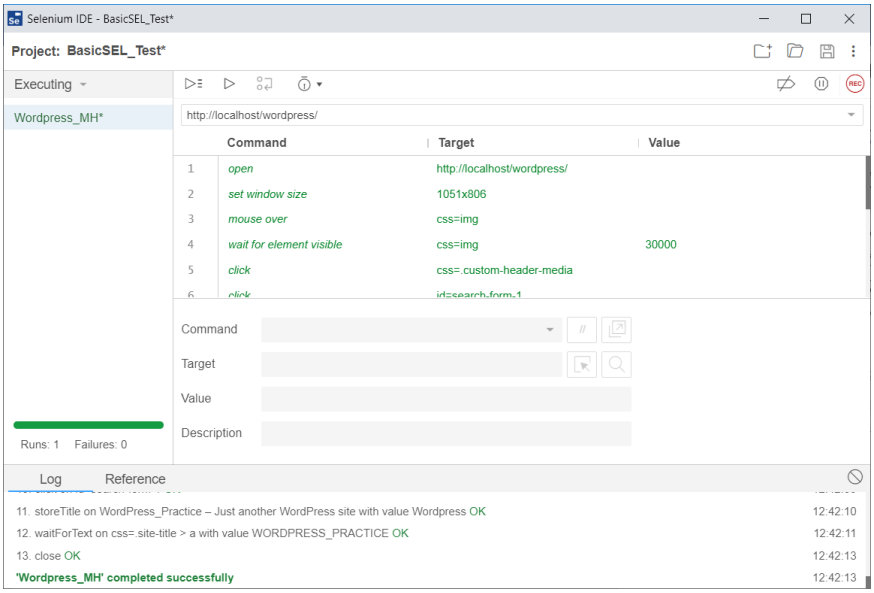


Test Case 3 :

`Test URL : http://localhost/wordpress/`

Command 1 -> wait for element visible
Command 2 -> verify text
Command 3 -> store title
Command 4 -> wait for text

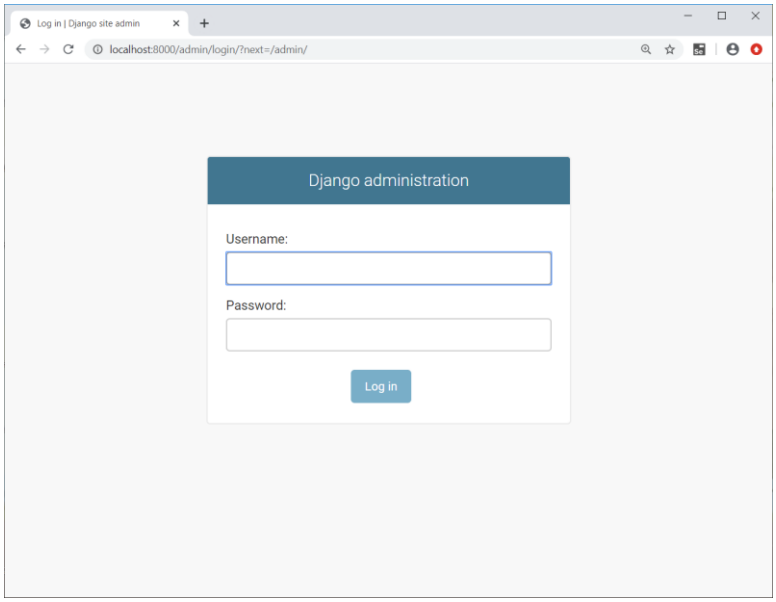


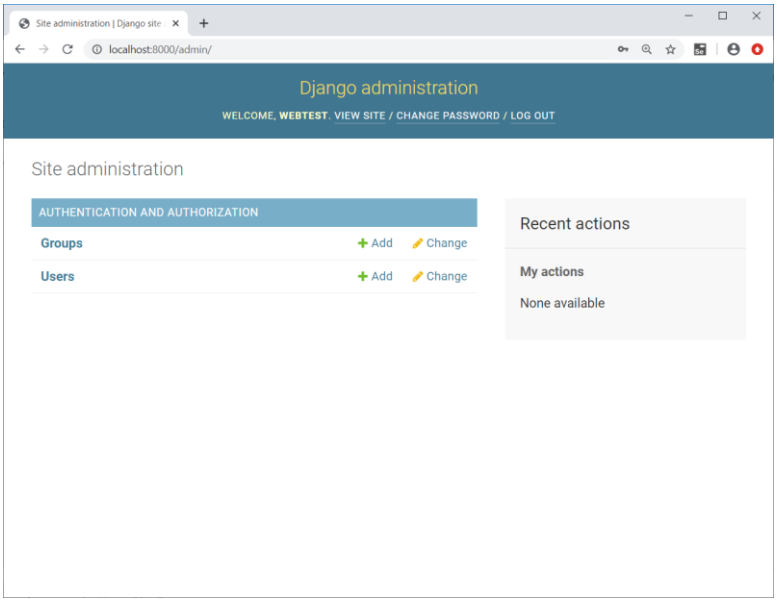


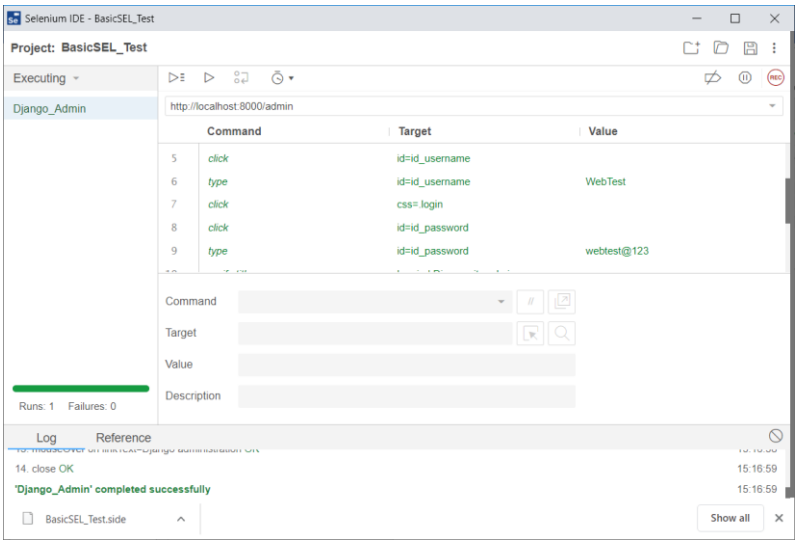
Test Case 4 :

`Test URL : http://localhost:8000/admin`

Command 1 -> Set Window Size.
Command 2 -> verify title
Command 3 -> assert not text
Command 4 -> mouse over


Log in | Django site admin
localhost:8000/admin/login/?next=/admin/
Django administration
Username:
Password:
Log in


Site administration | Django site
localhost:8000/admin/
Django administration
WELCOME, WEBTEST. VIEW SITE / CHANGE PASSWORD / LOG OUT
Site administration
AUTHENTICATION AND AUTHORIZATION
Groups + Add Change
Users + Add Change
Recent actions
My actions
None available


Selenium IDE - BasicSEL_Test
Project: BasicSEL_Test
Executing
Django_Admin
http://localhost:8000/admin
Command
Target
Value
5 click id=id_username
6 type id=id_username WebTest
7 click css=.login
8 click id=id_password
9 type id=id_password webtest@123
Command
Target
Value
Description
Runs: 1 Failures: 0
Log Reference
14. close OK 15:16:59
'Django_Admin' completed successfully 15:16:59
BasicSEL_Test.side
Show all