**Selenium Introduction:**

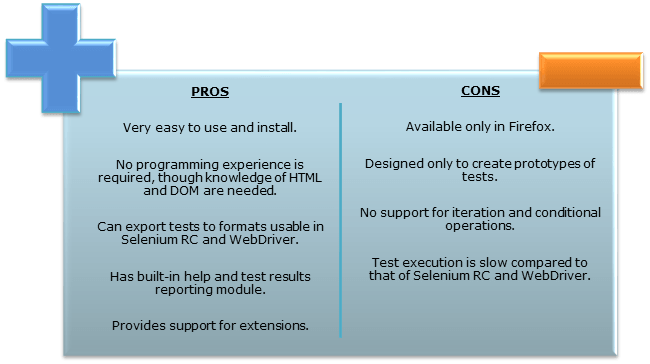
Selenium is an open source automated testing suite for web applications across different browsers and platforms. Selenium is not just a single tool but a suite of software's, each catering to different testing needs of an organization. **It has four components.**

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

[](https://cdn.guru99.com/images/SeleniumSuite.png)

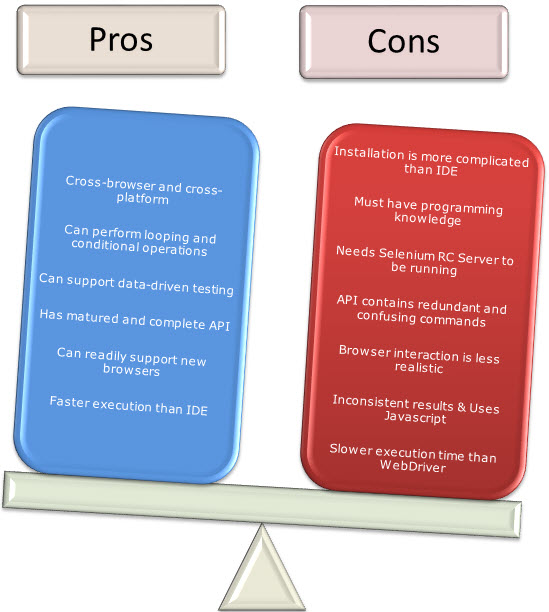
## Brief Introduction Selenium IDE

Selenium Integrated Development Environment (IDE) is the **simplest framework** in the Selenium suite and is **the easiest one to learn**. It is a **Firefox plugin** that you can install. Selenium IDE should only be used as a **prototyping tool**. If you want to create more advanced test cases, you will need to use either Selenium RC or WebDriver.

[](https://cdn.guru99.com/images/SeleniumIDEProCon.png)

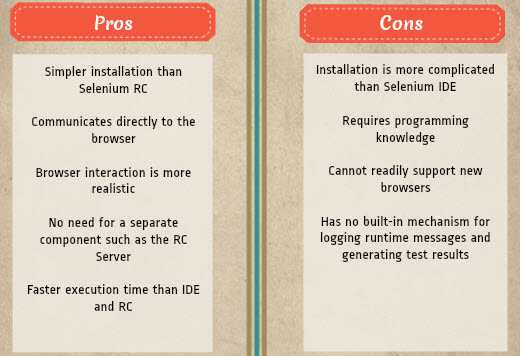
## Brief Introduction Selenium RC (Remote Controls)

Selenium RC was the **flagship testing framework** of the whole Selenium project for a long time. This is the first automated web testing tool that **allowed users to use a programming language they prefer** from the list : Java, C#, PHP, Python, Perl and Ruby.

[](https://cdn.guru99.com/images/RCProCon.jpg)

## Brief Introduction WebDriver

The WebDriver proves itself to be **better than both Selenium IDE and Selenium RC** in many aspects. WebDriver, unlike Selenium RC, does not rely on JavaScript for Automation. **It controls the browser by directly communicating with it. Supported languages:** Java, C#, PHP, Python, Perl and Ruby

[](https://cdn.guru99.com/images/pros_cons_-_webdriver.jpg)

## Selenium Grid

Selenium Grid is a tool **used together with Selenium RC to run parallel tests** across different machines and different browsers all at the same time.

**Features:**

* Enables **simultaneous running of tests** in **multiple browsers**
* **Saves execution time**enormously.
* Utilizes the **hub-and-nodes** concept. The hub acts as a central source of Selenium commands to each node connected to it.

**==================================================**

### Installation of Selenium IDE

1. Launch Firefox and navigate to <https://addons.mozilla.org/en-US/firefox/addon/selenium-ide/>. Click on Add to Firefox

## Installation of Firebug

**Firebug**is a Firefox add-on that we will use to **inspect the HTML elements** of the web application under test. It will provide us the name of the element that our Selenese command would act upon.

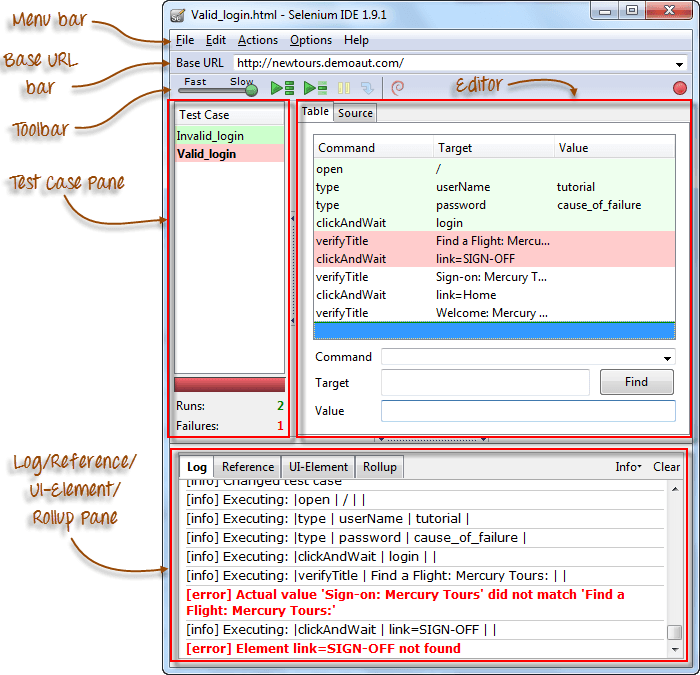
**1.** Use Firefox to navigate to Firebug's download page ( <https://addons.mozilla.org/en-US/firefox/addon/firebug/>) and click on the download link.

# Introduction to Selenium IDE

**Selenium IDE (Integrated Development Environment)** is the simplest tool in the Selenium Suite. It is a Firefox add-on that creates tests very quickly through its record-and-playback functionality. Though you will be able to use Selenium IDE without prior knowledge in programming, you should at least be familiar with HTML, JavaScript, and the DOM (Document Object Model) to utilize this tool to its full potential. Selenium IDE supports autocomplete mode when creating tests. This feature serves two purposes:

* It helps the tester to enter commands more quickly.
* It restricts the user from entering invalid commands.

## Features of Selenium IDE

[](https://cdn.guru99.com/images/ide_-_labelled_parts.png)

## Menu Bar

It is located at the **top most portion** of the IDE. The most commonly used menus are the File, Edit, and Options menus.

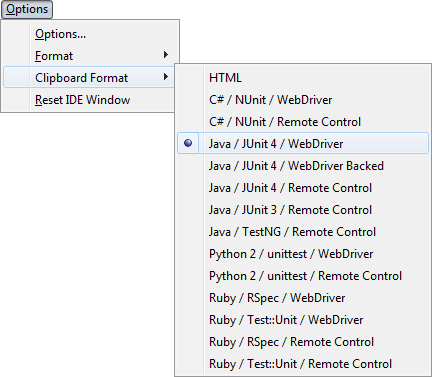
**File menu**

* It contains options to create, open, save and close tests.
* Tests are **saved in HTML format**.
* The most useful option is "**Export"** because **it allows you to turn your Selenium IDE test cases into file formats that can run on Selenium Remote Control and WebDriver**
* **"Export Test Case As..."**will export only the currently opened test case.
* **"Export Test Suite As..."**will export all the test cases in the currently opened test suite.

**Edit Menu**

* It contains usual options like Undo, Redo, Cut, Copy, Paste, Delete, and Select All.
* The two most important options are the "**Insert New Command**" and "**Insert New Comment**".
* The newly inserted command or comment **will be placed on top of the currently selected line**.
* **Commands**are colored **black**.
* **Comments**are colored **purple.**

## ****Options menu****

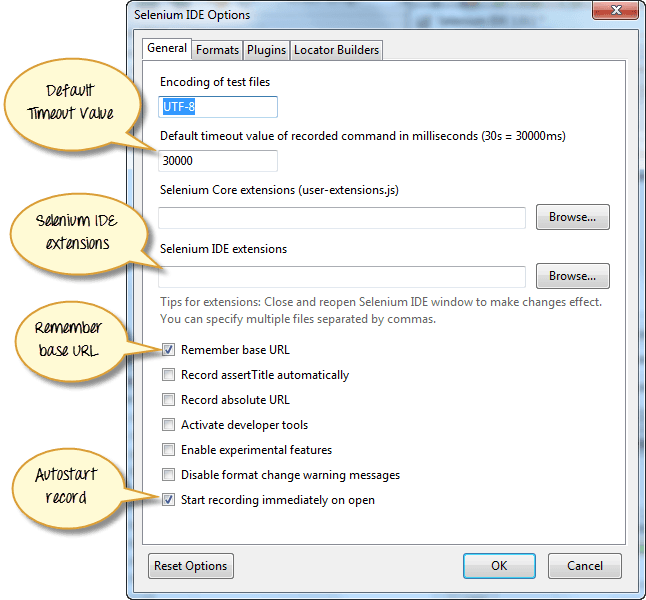
[](https://cdn.guru99.com/images/clipboard_format_menu_list_complete.png)It provides the **interface for configuring various settings** of Selenium IDE. We shall concentrate on the **Options** and **Clipboard Format** options.

|  |  |
| --- | --- |
|  |  |

**Clipboard Format**

* The clipboard format allows you to copy a Selenese command from the editor and paste it as a code snippet.
* The format of the code follows the option selected in the clipboard format's list.
* HTML is the default selection.

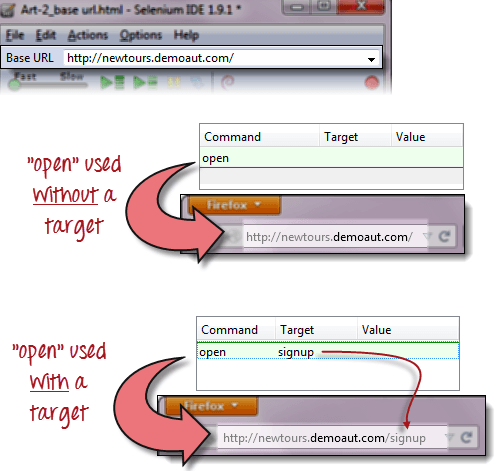
## Selenium IDE Options dialog box

You can launch the Selenium IDE Options dialog box by clicking Options > Options... on the menu bar. [](https://cdn.guru99.com/images/options_dialog_with_lables.png)

* **Default Timeout Value**. This refers to the time that Selenium has to wait for a certain element to appear or become accessible before it generates an error.  **Default timeout value is 30000ms**.
* **Selenium IDE extensions**. This is where you specify the extensions you want to use to extend Selenium IDE's capabilities. You can visit <http://addons.mozilla.org/en-US/firefox/>and use "Selenium" as a keyword to search for the specific extensions.
* **Remember base URL.**Keep this checked if you want Selenium IDE to remember the Base URL every time you launch it. If you uncheck this, Selenium IDE will always launch with a blank value for the Base URL.
* **Autostart record.** If you check this, Selenium IDE will immediately record your browser actions upon startup.
* **Locator builders.**This is where you specify the order by which locators are generated while recording. **Locators are ways to tell Selenium IDE which UI element should a Selenese command act upon**. In the setup below, when you click on an element with an ID attribute, that element's ID will be used as the locator since "id" is the first one in the list. If that element does not have an ID attribute, Selenium will next look for the "name" attribute since it is second in the list. The list goes on and on until an appropriate one is found.

## Base URL Bar

* It has **a dropdown menu that** **remembers all previous values** for easy access.
* The Selenese command **"open" will take you to the URL that you specified in the Base URL**.
* **The Base URL is very useful in accessing relative URLs**. Suppose that your Base URL is set to [http://newtours.demoaut.com](http://newtours.demoaut.com/). When you execute the command "open" with the target value "signup," Selenium IDE will direct the browser to the sign-up page. See the illustration below.

[](https://cdn.guru99.com/images/BaseURLTarget.png)

**Toolbar**

|  |  |
| --- | --- |
| [Introduction to Selenium IDE](https://cdn.guru99.com/images/fast-slow.png) | **Playback Speed**. This controls the speed of your Test Script Execution. |
| [Introduction to Selenium IDE](https://cdn.guru99.com/images/record.png) | **Record.**This starts/ends your recording session. ach browser action is  entered as a Selenese command in the Editor. |
| [Introduction to Selenium IDE](https://cdn.guru99.com/images/play_entire_test_suite.png) | **Play entire test suite**. This will sequentially play all the test cases listed  in the[Test Case](https://www.guru99.com/test-case.html)Pane. |
| [Introduction to Selenium IDE](https://cdn.guru99.com/images/play_current_test_case.png) | **Play current test case**. This will play only the currently selected test case  in the Test Case Pane. |
| [Introduction to Selenium IDE](https://cdn.guru99.com/images/pause_resume_2.png) | **Pause/Resume**. This will pause or resume your playback. |
| [Introduction to Selenium IDE](https://cdn.guru99.com/images/step.png) | **Step**. This button will allow you to step into each command in your test script. |
| [Introduction to Selenium IDE](https://cdn.guru99.com/images/rollup_button.png) | **Apply rollup rules**. This is an advanced functionality. It allows you to group Selenese commands together and execute them as a single action. |

## Test Case Pane

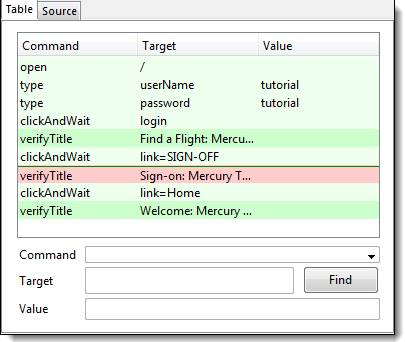
|  |  |
| --- | --- |
| [Introduction to Selenium IDE](https://cdn.guru99.com/images/test_case_pane_-_final.png) | * In Selenium IDE, you can open **more than one test case at a time**. * **The test case pane shows you the list of currently opened test cases.** * When you open a test suite, the test case pane will **automatically list all the test cases** contained in it. * The test case written in **bold font** is the **currently selected test case** * After playback, **each test case is color-coded** to represent if it passed or failed.   + Green color means "Passed."   + Red color means "Failed." * At the bottom portion is a summary of the number of test cases that were run and failed. |

## Editor

You can think of the editor as **the place where all the action happens**. It is available in two views: Table and Source.

##### Table View

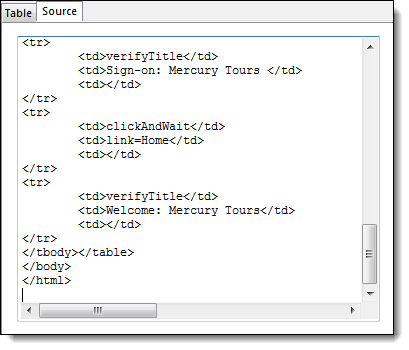
* Most of the time, you will work on Selenium IDE using the **Table View**.
* This is **where you create and modify Selenese commands.**
* After playback, each step is color-coded.

[](https://cdn.guru99.com/images/Editor.png)

* To create steps, type the name of the command in the "Command" text box. **It displays a dropdown list of commands**that match with the entry that you are currently typing.
* Target is any parameter (like username, password) for a command and Value is the input value (like tom, 123pass) for those Targets.

##### Source View

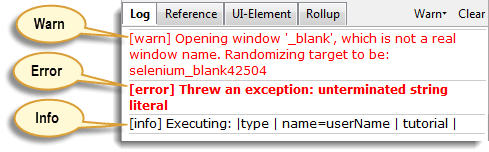
* It displays the steps in HTML (default) format.
* It also allows you to edit your script just like in the Table View.

[](https://cdn.guru99.com/images/editor_source_-_final.png)

## Log Pane

**The Log Pane displays runtime messages**during execution. It provides real-time updates as to what Selenium IDE is doing. **Logs are categorized into four types:**

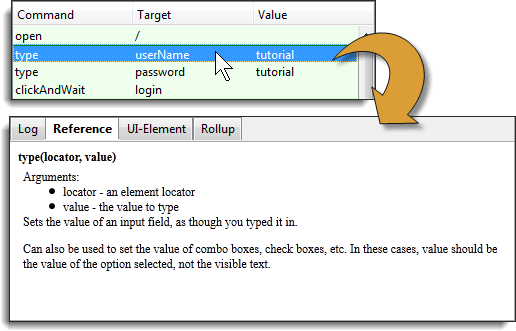
* Debug - By default, Debug messages are not displayed in the log panel. They show up only when you filter them. They provide technical information about what Selenium IDE is doing behind the scenes. It may display messages such as a specific module has done loading, a certain function is called, or an external JavaScript file was loaded as an extension.
* Info - It says which command Selenium IDE is currently executing.
* Warn - These are warning messages that are encountered in special situations.
* Error - These are error messages generated when Selenium IDE fails to execute a command, or if a condition specified by "verify" or "assert" command is not met.

[](https://cdn.guru99.com/images/Log_pane_-_types_of_logs.png)

## 

## Reference Pane

**The Reference Pane shows a concise description of the currently selected Selenese command in the Editor**. It also shows the **description about the locator and value** to be used on that command.

[](https://cdn.guru99.com/images/complete.png)

**==================================================**