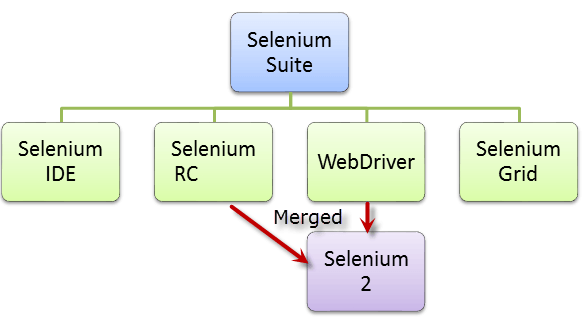
363201500124 ifsc ICIC0003632 What is selenium

Selenium is an open source and a portable automated software testing tool for testing web applications It has capabilities to operate across different browsers and operating systems. Selenium is not just a single tool but a set of tools that helps testers to automate webbased applications more efficiently.

**Selenium is an open-source tool that is used for test automation. It is licensed under Apache License 2.0. Selenium is a suite of tools that helps in automating only web applications**

**It has 4 components.**

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

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

* + **Selenium IDE:** This is in fact an add-on feature for Mozilla Firefox. This is generally used for playback, record, debugging and edit of selenium scripts.
  + **Selenium RC:** This is alternatively known as Selenium Remote Control and is utilized to run tests across various systems and browsers.
  + **Selenium Grid:** It is capable of running diverse cases of Selenium RC at the same time. Selenium Grid facilitates distribution of tests across different browsers and machines so that parallel execution of the tests can be carried out.
  + **Selenium Webdriver:** This is the latest version of Selenium which comprises a number of Selenium APIs.

|  |  |
| --- | --- |
| HP UFT (QTP) | Selenium |
| It is commercial tool and hence it requires a license is expensive | Its an open source tool hence it does not require license and is free |
| It is used for testing client-server applications. It can test web- based as well as desktop applications | Using Selenium only web applications can be tested |
| QTP  tests can only be developed in QTP IDE | Selenium has the option of using wide range of IDEs like Visual Studio, Eclipse, Netbeans |
| HP UFT only supports VB script | Selenium supports JAVA, .NET, Ruby, Perl, PHP, and many other programming languages |
| HP UFT comes with built in object repository.  Object repository development and maintenance is quite easy in HP ALM | Selenium dose not  have such built in object repository, but object can be managed by using UI element user extension |
| HP UFT integrates with test management tool like HP Quality Center | There is no such tool that integrates with Selenium |
| All types of dialog box is supported by it | It supports dialog box partially |
| It supports different environments like SAP, Oracle, .NET  but user need to purchase add-on license for them | It supports addition of plug-ins to include features that are not provided by core Selenium |
| Automation testing is expensive due to licensing cost | Web automation testing with Selenium incurs less cost |
| HP QTP offers very good technical support | It has no official support, since it is an open source |
| HP QTP supports only VBScript . Hence Object Oriented Approach to Test Automation becomes challenging | Selenium supports mature Object Oriented Language like Java |
| HP QTP is more user friendly and scripts are developed quickly. | It&rsquos a low level tool with less functionality. Script requires more time to develop and maintain |
| Backward compatibility is powerful. Latest version of HP ALM supports code that was developed 5 years back | With new Selenium release the API changes. So Test Scripts need to be updated |

**Advantages**

1. Selenium is pure open source, freeware and portable tool.   
2. Selenium supports variety of languages that include Java, Perl, Python, C#, Ruby, Groovy, Java Script, and VB Script. etc.   
3. Selenium supports many operating systems like Windows, Macintosh, Linux, Unix etc.   
4. Selenium supports many browsers like Internet explorer, Chrome, Firefox, Opera, Safari etc.   
5. Selenium can be integrated with ANT or Maven kind of framework for source code compilation.   
6. Selenium can be integrated with TestNG testing framework for testing our applications and generating reports.   
7. Selenium can be integrated with Jenkins or Hudson for continuous integration.   
8. Selenium can be integrated with other open source tools for supporting other features.   
9. Selenium can be used for Android, IPhone, Blackberry etc. based application testing.   
10. Selenium supports very less CPU and RAM consumption for script execution.   
11. Selenium comes with different component to provide support to its parent which is Selenium IDE, Selenium Grid and Selenium Remote Control (RC).

**Disadvantages:**   
  
1. Selenium needs very much expertise resources. The resource should also be very well versed in framework architecture.   
2. Selenium only supports web based application and does not support windows based application.   
3. It is difficult to test Image based application.   
4. Selenium need outside support for report generation activity like dependence on TestNG or Jenkins.   
5. Selenium does not support built in add-ins support.   
6. Selenium user lacks online support for the problems they face.   
7. Selenium does not provide any built in IDE for script generation and it need other IDE like Eclipse for writing scripts.   
8. Selenium Automation Engineers are bit in scarcity these days.   
9. Selenium script creation time is bit high.   
10. Selenium does not support file upload facility.   
11. Selenium partially supports for Dialog boxes.

**What is dom**

Xpath locators and css selectors are used in the context of **selenium** web driver, and **DOM** locators are used in the context of javascript (i.e. to locate element with **DOM** locator properly you shoulda wrap **DOM** locators with JavascriptExecutor firstly).

**Selenium ide**

**Selenium IDE** is an integrated development environment for **Selenium** scripts. It is implemented as a Firefox extension, and allows you to record, edit, and debug tests.

Selenium IDE stands for Selenium Integrated Development Environment. It is a Firefox plugin which helps testers to recorded their actions and then export them as a reusable script in one of many programming languages. Selenium IDE has a recording feature, that provides an easy-to-use interface for developing automated tests.

Selenium IDE was developed by Shinya Kasatani. Selenium IDE became a part of Selenium Package in the year 2006. Then tool becomes a great value and potential to the community.

Selenium IDE is the most simplest and easiest of all the tools within the Selenium Package. The record and playback feature makes it exceptionally easy to learn with minimal knowledge of any programming language. The disadvantages of IDE are in reality not disadvantages, but just limitations to what the IDE could achieve. These limitations can be overcome by using Selenium RC or WebDriver.

**Advantages**

* Selenium IDE is very easy to use.
* It has the capability to convert the test to different programming languages such as html, java etc
* Programming language experience is not required for Selenium IDE
* Selenium IDE provides Logging capabilities using file login plug-in.
* In Selenium IDE, user can debug and set breakpoints
* Selenium IDE is flexible for the users.

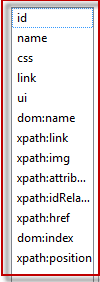
**Disadvantage**

* Selenium IDE is Firefox plugin, thus its support is limited to Firefox only
* It will not support iteration and conditional statement
* Selenium IDE doesn't support error handling
* Doesn't support test script grouping
* Selenium IDE do not support Database testing

Hope this article helps you to understand advantages and disadvantages of selenium IDE!!

# Selenium IDE Element Locator

#### Selenium has several ways to identify web elements when using Selenium IDE or Selenium API in .Net or Java programming. Selenium can locate elements:

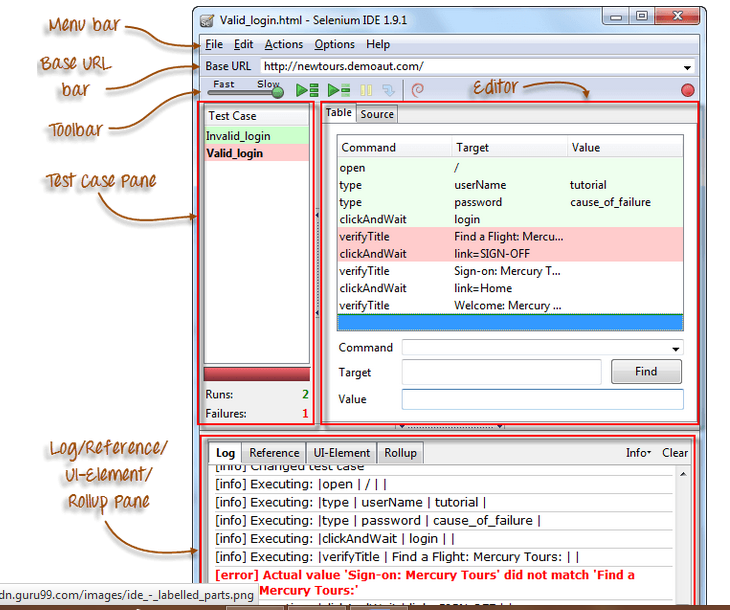


* by ID
* by name
* by link
* by xpath
* by CSS
* by dom=document.getElementById("ElementID");

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

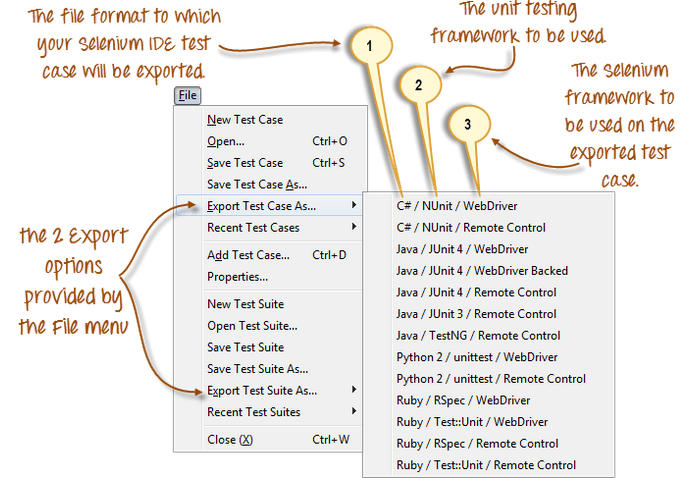


### Menu Bar

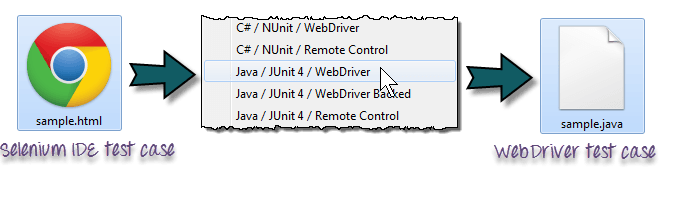
It is located at the **top most portion** of the IDE. The most commonly used menus are the File, Edit, and Options menus.

**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.



* As of **Selenium IDE v1.9.1**, test cases can be exported only to the following formats:
* .cs (C# source code)
* .java (Java source code)
* .py (Python source code)
* .rb (Ruby source code)



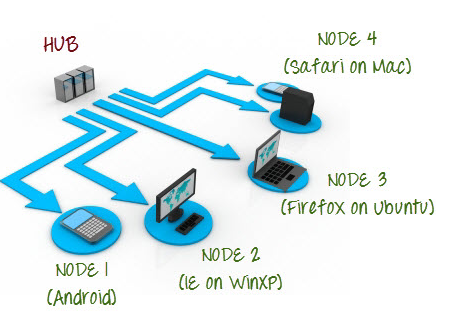
Applicant programming interface

|  |  |  |
| --- | --- | --- |
| **Selenium IDE** | **Selenium RC** | **Selenium WebDriver** |
| It only works in Mozilla browser. | It supports with all browsers like Firefox, IE, Chrome, Safari, Opera etc. | It supports with all browsers like Firefox, IE, Chrome, Safari, Opera etc. |
| It supports Record and playback | It doesn’t supports Record and playback | It doesn’t supports Record and playback |
| Doesn’t required to start server before executing the test script. | Required to start server before executing the test script. | Doesn’t required to start server before executing the test script. |
| It is a GUI Plug-in | It is standalone java program which allow you to run Html test suites. | It actual core API which has binding in a range of languages. |
| Core engine is Javascript based | Core engine is Javascript based | Interacts natively with browser application |
| Very simple to use as it is record & playback. | It is easy and small API | As compared to RC, it is bit complex and large API. |
| It is not object oriented | API’s are less Object oriented | API’s are entirely Object oriented |
| It doesn’t supports of moving mouse cursors. | It doesn’t supports of moving mouse cursors. | It supports of moving mouse cursors. |
| Need to append full xpath with ‘xpath=\\’ syntax | Need to append full xpath with ‘xpath=\\’ syntax | No need to append full xpath with ‘xpath=\\’ syntax |
| It does not supports listeners | It does not supports listeners | It supports the implementation of listeners |
| It does not support to test iphone/Android applications | It does not support to test iphone/Android applications | It support to test iphone/Android applications |

## What is Selenium Grid?

**Selenium Grid is a part of the Selenium Suite that specializes on running multiple tests across different browsers, operating systems, and machines in parallel**.

Selenium Grid has 2 versions - the older Grid 1 and the newer Grid 2. We will only focus on Grid 2 because Grid 1 is gradually being deprecated by the Selenium Team.



Selenium Grid uses a hub-node concept where you only run the test on a single machine called a **hub**, but the execution will be done by different machines called **nodes**.

## When to Use Selenium Grid?

You should use Selenium Grid when you want to do either one or both of following:

* **Run your tests against different browsers, operating systems, and machines all at the same time.**This will ensure that the application you are testing is fully compatible with a wide range of browser-O.S combinations.
* **Save time in the execution of your test suites**. If you set up Selenium Grid to run, say, 4 tests at a time, then you would be able to finish the whole suite around 4 times faster.

## Grid 1.0 Vs Grid 2.0

Following are the main differences between Selenium Grid 1 and 2.

|  |  |
| --- | --- |
| **Grid 1** | **Grid 2** |
| Selenium Grid 1 has its own remote control that is different from the Selenium RC server. They are two different programs. | Selenium Grid 2 is now bundled with the Selenium Server jar file |
| You need to install and configure Apache Ant first before you can use Grid 1. | You do not need to install Apache Ant in Grid 2. |
| Can only support Selenium RC commands/scripts. | Can support both Selenium RC and WebDriver scripts. |
| You can only automate one browser per remote control. | One remote control can automate up to 5 browsers. |

## What is a Hub and Node?

### The Hub

* The hub is the central point where you load your tests into.
* There should only be one hub in a grid.
* The hub is launched only on a single machine, say, a computer whose O.S is Windows 7 and whose browser is IE.
* The machine containing the hub is where the tests will be run, but you will see the browser being automated on the node.

### The Nodes

* Nodes are the Selenium instances that will execute the tests that you loaded on the hub.
* There can be one or more nodes in a grid.
* Nodes can be launched on multiple machines with different platforms and browsers.
* The machines running the nodes need not be the same platform as that of the hub.

|  |
| --- |
| class sathish{  public static void main(String args[]){  System.out.println("Hello sathish");  }  } |

In the above application example we are using public static void main. Each word has a different meaning and purpose.

**Public** : is an [**Access Modifier**](http://javabeginnerstutorial.com/core-java-tutorial/access-modifier-in-java/), which defines who can access this Method. Public means that this Method will be accessible by any Class(If other Classes are able to access this Class.).

**Static :**is a keyword which identifies the class related thing. This means the given Method or variable is not instance related but Class related. It can be accessed without creating the instance of a Class.

**Void :** is used to define the Return Type of the Method. It defines what the method can return. Void means the Method will not return any value.

**main:**is the **n**ame of the Method**.**This Method name is searched by JVM as a starting point for an application with a particular signature only.

**String args[] :** is the parameter to the main Method.

**CLASS; it is a group of data variables**

**Method : methods are two types predefined and user defined**