



Selenium Guía rápida

[◀ Anterior \(selenium_grids.html\)](#)[Siguiendo capítulo ▶ \(selenium_useful_resources.html\)](#)

El selenio referencia rápida

Selenio - Introducción

El selenio es un código abierto y una herramienta de pruebas de software automatizado portátil para probar aplicaciones web. Tiene capacidad para operar a través de diferentes navegadores y sistemas operativos. El selenio no es sólo una herramienta única, sino un conjunto de herramientas que ayuda a los probadores para automatizar aplicaciones basadas en la Web de forma más eficiente.

Veamos ahora entendemos cada una de las herramientas disponibles en la suite de selenio y su uso.

Herramienta	Descripción
selenio IDE	El selenio que NTEGRADO D esarrollo mbiente (IDE) es un plugin para Firefox que permite a los probadores para registrar sus acciones, ya que siguen el flujo de trabajo que tienen que realizar la prueba.



El selenio RC	El selenio Remote Control (RC) fue el marco de pruebas insignia que permitió a más de acciones del navegador simples y ejecución lineal. Se hace uso de toda la potencia de los lenguajes de programación como Java, C #, PHP, Python, Ruby y Perl para crear pruebas más complejas.
WebDriver selenio	WebDriver El selenio es el sucesor de selenio RC que envía comandos directamente en el navegador y recupera los resultados.
El selenio cuadrícula	Rejilla de selenio es una herramienta que se utiliza para ejecutar las pruebas paralelas a través de diferentes máquinas y diferentes navegadores al mismo tiempo que se traduce en tiempo de ejecución reduce al mínimo.

Ventajas de selenio

QTP y el selenio son las herramientas más utilizadas en el mercado de automatización de pruebas de software. Por lo tanto, tiene sentido comparar los pros de selenio sobre QTP.

Selenio	QTP
El selenio es una herramienta de código abierto.	QTP es una herramienta comercial y hay un costo involucrado en cada una de las licencias.
Puede ser extendido por diversas tecnologías que exponen DOM.	complementos y necesidades limitadas complementos para cada una de las tecnologías.
Tiene capacidad para ejecutar secuencias de comandos a través de diferentes navegadores.	Puede ejecutar las pruebas en las versiones específicas de Firefox, IE y Chrome.
Puede ejecutar secuencias de comandos en varios sistemas operativos.	Sólo funciona con Windows.
Es compatible con los dispositivos móviles.	Es compatible con los dispositivos móviles con la ayuda de herramientas de terceros.
Ejecuta las pruebas dentro del navegador, por lo que no se requiere foco mientras que la ejecución del script está en curso.	Necesidades de enfoque durante la ejecución del script, como la herramienta actúa sobre las acciones del navegador (imita el usuario).



Se puede ejecutar pruebas en paralelo con el uso de selenio Grids.

QTP no puede ejecutar pruebas en paralelo, sin embargo, la integración de QTP con control de calidad permite a los probadores para ejecutar en paralelo. Control de calidad es también una herramienta comercial.

Desventajas de selenio

Ahora vamos a discutir los peligros de selenio sobre QTP.

Selenio	QTP
Soporta aplicaciones basadas en web solamente.	Puede probar las aplicaciones web y de escritorio.
Sin Características tales como repositorio de objetos / escenario de recuperación	QTP se ha incorporado en los repositorios de objetos y escenarios de recuperación.
Sin IDE, por lo que el desarrollo del guión no será tan rápido como QTP.	IDE más intuitivo; automatización se puede lograr más rápidamente.
No se puede acceder a los controles dentro del navegador.	Se puede acceder a los controles en el navegador como barra de favoritos, hacia atrás y botones de avance.
Sin la generación de informes de prueba predeterminado.	la generación de resultados de la prueba por defecto dentro de la herramienta.
Para la parametrización, los usuarios tienen que confiar en el lenguaje de programación.	La parametrización es incorporada y fácil de implementar.

Selenio - IDE

El selenio-IDE

El selenio-IDE (Integrated Development Environment) es un complemento de Firefox fácil de usar para el desarrollo de casos de prueba de selenio. Proporciona una interfaz gráfica de usuario para el registro de las acciones del usuario utilizando Firefox que se utiliza para aprender y utilizar selenio, pero sólo se puede utilizar con el navegador Firefox como no son compatibles con otros navegadores.

Sin embargo, los scripts grabados se pueden convertir en varios lenguajes de programación soportados por Selenium y las secuencias de comandos se pueden ejecutar en otros navegadores también.

Título	Descripción
Descargar Selenium IDE (selenium_download_ide.html)	Esta sección se ocupa de cómo descargar y configurar Selenium IDE.
Características IDE selenium (selenium_ide_tool_features.html)	Esta sección se ocupa de las funciones disponibles en Selenium IDE.
Creación de pruebas Selenium IDE (selenium_ide_test_creation.html)	Esta sección se ocupa de cómo crear pruebas IDE usando la función de grabación.
Selenium IDE depuración de scripts (selenium_ide_debugging.html)	Esta sección se ocupa de la depuración de la escritura de selenium IDE.
Inserción de puntos de verificación (selenium_ide_verification_points.html)	En esta sección se describe cómo insertar puntos de verificación en Selenium IDE.
El selenium coincidencia de patrones (selenium_ide_pattern_matching.html)	Esta sección se ocupa de cómo trabajar con expresiones regulares usando IDE.
Extensiones de selenium de los usuarios (selenium_user_extensions.html)	El script de Java que permite a los usuarios personalizar y añadir nueva funcionalidad.
Ejecución del navegador diferente (selenium_ide_on_different_browser.html)	Esta sección se ocupa de la forma de ejecutar secuencias de comandos IDE selenium en diferentes navegadores.

Selenium - Configuración del entorno


(/ ~ \)

Selenium Tutorial

Selenium Inicio
(/es/selenium/default.html)

Selenium Visión de conjunto
(/es/selenium/selenium_overview.html)

Selenium - Configuración del entorno

Para el desarrollo de selenium RC o guiones WebDriver, los usuarios tienen que asegurarse de que han hecho la configuración inicial. Configuración del entorno implica los siguientes pasos.

- **Descargar e instalar Java**
- **Descargar y configurar Eclipse**
- **Configurar quemador, y la FirePath**
- **Configurar Selenium RC**
- **El selenium configurar WebDriver**

http://www.w3ii.com/es/selenium/selenium_quick_guide.html

4/80

Selenium IDE

(/es/selenium/selenium_ide)

Selenium Configuración
del entorno

(/es/selenium/selenium_en)

Selenium Control remoto

(/es/selenium/selenium_rc.l)

Selenium Comandos

Selenese

(/es/selenium/selenium_sel)

Selenium WebDriver

(/es/selenium/selenium_we)

Selenium localizadores

(/es/selenium/selenium_loc)

Selenium Las

interacciones de los
usuarios

(/es/selenium/selenium_use)

Selenium Técnicas de
diseño de pruebas

(/es/selenium/selenium_tes)

Selenium TestNG

(/es/selenium/selenium_tes)

Selenium rejillas

(/es/selenium/selenium_gri)

Selenium Guía rápida

(/es/selenium/selenium_qu)

Selenium Recursos útiles

(/es/selenium/selenium_use)

Descargar e instalar Java

Necesitamos tener el JDK (Java Development Kit) instalado con el fin de trabajar con Selenium WebDriver / selenium. Veamos cómo descargar e instalar Java.

Paso 1: Vaya a la URL:

<http://www.oracle.com/technetwork/java/javase/downloads/index.html>

Paso 2: Vaya a la sección "Descargas" y seleccionar "Descargar JDK".

Overview **Downloads** Documentation Community Technologies Training

Java SE Downloads


DOWNLOAD


DOWNLOAD

Java Platform (JDK) 8u11 JDK 8u11 & NetBeans 8.0

Java Platform, Standard Edition

Java SE 8u11
This release includes important security fixes. Oracle strongly recommends that all Java SE 8 users upgrade to this release.
[Learn more](#)

- Installation Instructions
- Release Notes
- Oracle License
- Java SE Products
- Third Party Licenses
- Certified System Configurations
- Readme Files
 - JDK ReadMe
 - JRE ReadMe

JDK
DOWNLOAD

Server JRE
DOWNLOAD

JRE
DOWNLOAD

Paso 3: Seleccione "Aceptar el acuerdo de licencia" botón de radio.



Java SE Development Kit 8 Downloads

Thank you for downloading this release of the Java™ Platform, Standard Edition Development Kit (JDK™). The JDK is a development environment for building applications, applets, and components using the Java programming language.

The JDK includes tools useful for developing and testing programs written in the Java programming language and running on the Java platform.

See also:

- Java Developer Newsletter (tick the checkbox under Subscription Center > Oracle Technology News)
- Java Developer Day hands-on workshops (free) and other events
- Java Magazine

JDK MD5 Checksum

Looking for JDK 8 on ARM?

JDK 8 for ARM downloads have moved to the [JDK 8 for ARM download page](#).

Java SE Development Kit 8u11

You must accept the [Oracle Binary Code License Agreement for Java SE](#) to download this software.

☒ Accept License Agreement ☐ Decline License Agreement

Product / File Description	File Size	Download
Linux x86	133.58 MB	jdk-8u11-linux-i586.rpm
Linux x86	152.55 MB	jdk-8u11-linux-i586.tar.gz
Linux x64	133.89 MB	jdk-8u11-linux-x64.rpm
Linux x64	151.65 MB	jdk-8u11-linux-x64.tar.gz
Mac OS X x64	207.82 MB	jdk-8u11-macosx-x64.dmg
Solaris SPARC 64-bit (SVR4 package)	135.66 MB	jdk-8u11-solaris-sparcv9.tar.Z
Solaris SPARC 64-bit	96.14 MB	jdk-8u11-solaris-sparcv9.tar.gz
Solaris x64 (SVR4 package)	135.7 MB	jdk-8u11-solaris-x64.tar.Z
Solaris x64	93.18 MB	jdk-8u11-solaris-x64.tar.gz
Windows x86	151.81 MB	jdk-8u11-windows-i586.exe
Windows x64	155.29 MB	jdk-8u11-windows-x64.exe

Paso 4: Seleccione la instalación apropiada. En este caso, es poco 'de Windows 7-64'. Haga clic en el enlace correspondiente y guarde el archivo .exe en el disco.

Java SE Development Kit 8u11

You must accept the [Oracle Binary Code License Agreement for Java SE](#) to download this software.

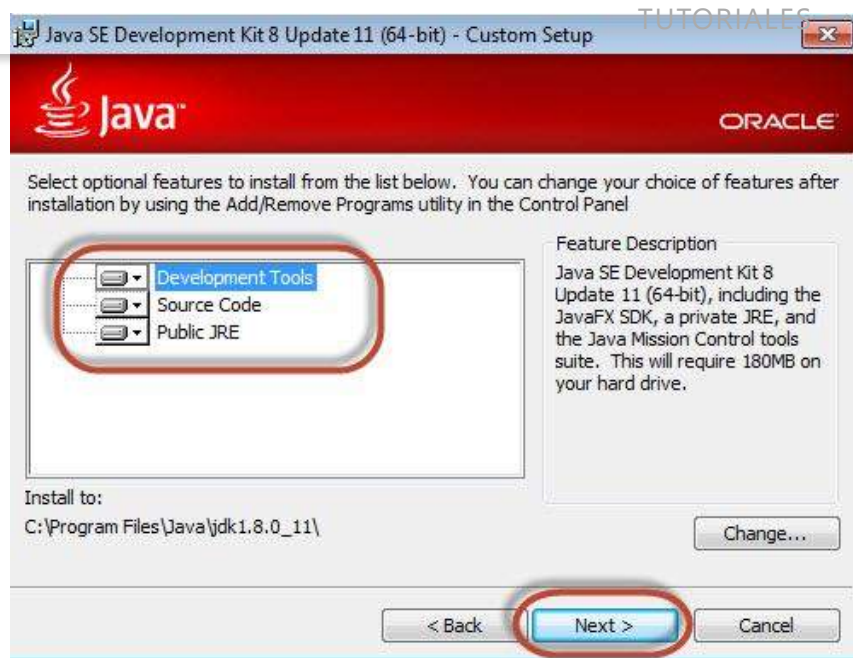
Thank you for accepting the Oracle Binary Code License Agreement for Java SE; you may now download this software.

Product / File Description	File Size	Download
Linux x86	133.58 MB	jdk-8u11-linux-i586.rpm
Linux x86	152.55 MB	jdk-8u11-linux-i586.tar.gz
Linux x64	133.89 MB	jdk-8u11-linux-x64.rpm
Linux x64	151.65 MB	jdk-8u11-linux-x64.tar.gz
Mac OS X x64	207.82 MB	jdk-8u11-macosx-x64.dmg
Solaris SPARC 64-bit (SVR4 package)	135.66 MB	jdk-8u11-solaris-sparcv9.tar.Z
Solaris SPARC 64-bit	96.14 MB	jdk-8u11-solaris-sparcv9.tar.gz
Solaris x64 (SVR4 package)	135.7 MB	jdk-8u11-solaris-x64.tar.Z
Solaris x64	93.18 MB	jdk-8u11-solaris-x64.tar.gz
Windows x86	151.81 MB	jdk-8u11-windows-i586.exe
Windows x64	155.29 MB	jdk-8u11-windows-x64.exe

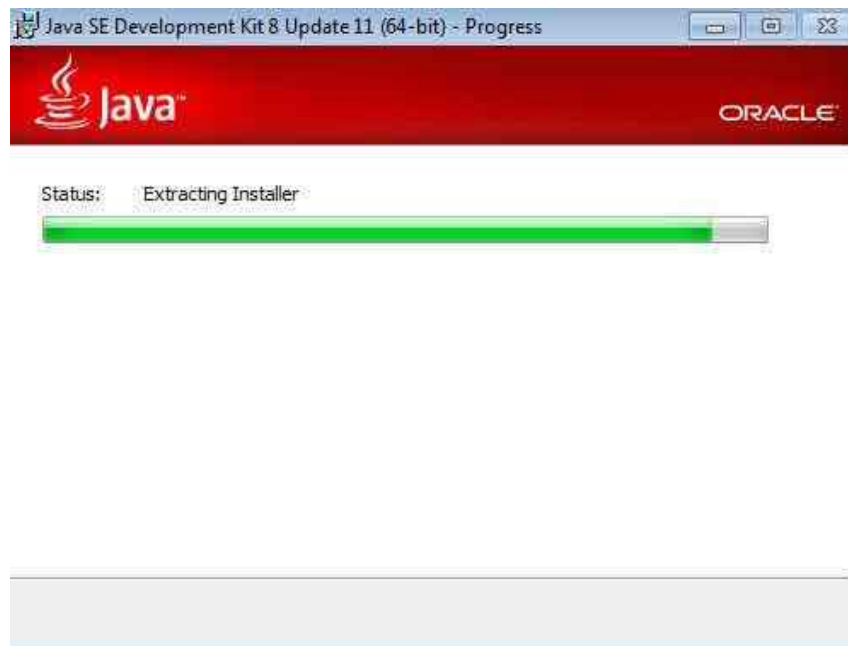
Paso 5: Ejecutar el archivo ejecutable descargado para iniciar el asistente de instalación. Haga clic en "Siguiente" para continuar.



Paso 6: Selección de las funciones y haga clic en "Siguiente".



Paso 7: El programa de instalación se extrae y se muestra su progreso en el asistente.

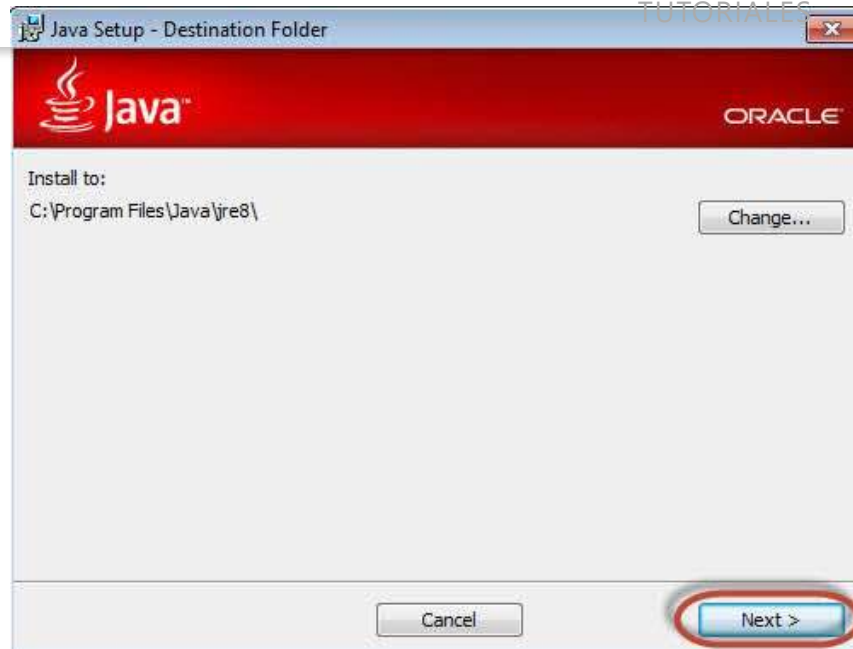


Paso 8: El usuario puede elegir la ubicación de instalación y haga clic en "Siguiendo".



TUTORIALES

Español ▼



Paso 9: El instalador instala el JDK y los nuevos archivos se copian al otro lado.



Paso 10: El instalador instala correctamente y muestra la misma para el usuario.



Español ▼

Paso 11: Para verificar si la instalación se ha realizado correctamente, vaya a la línea de comandos y sólo tiene que escribir 'java' como un comando. La salida del comando se muestra a continuación. Si la instalación de Java no se realiza correctamente o si no hubiera sido instalado, se lanzaría un error "comando desconocido".

```
C:\Windows\system32\cmd.exe
C:\>java
Usage: java [-options] class [args...]
        (to execute a class)
or java [-options] -jar jarfile [args...]
        (to execute a jar file)
where options include:
-d32          use a 32-bit data model if available
-d64          use a 64-bit data model if available
-server      to select the "server" VM
              The default VM is server.

-cp <class search path of directories and zip/jar files>
-classpath <class search path of directories and zip/jar files>
           A ; separated list of directories, JAR archives,
           and ZIP archives to search for class files.
-D<name>=<value>  set a system property
-verbose[:<classname>] enable verbose output
-version        print product version and exit
-version:<value> require the specified version to run
-showversion    print product version and continue
-jre-restrict-search | -no-jre-restrict-search
                include/exclude user private JREs in the version search
-? -help       print this help message
-X             print help on non-standard options
-eaf:<packagename>...![:<classname>]
-enableassertions[:<packagename>...![:<classname>]]
                enable assertions with specified granularity
-daf:<packagename>...![:<classname>]
-disableassertions[:<packagename>...![:<classname>]]
                disable assertions with specified granularity
-esa | -enablesystemassertions
                enable system assertions
-dsa | -disablesystemassertions
                disable system assertions
-agentlib:<libname>[=<options>]
                load native agent library <libname>, e.g. -agentlib:hprof
                see also, -agentlib:jdwp=help and -agentlib:hprof=help
-agentpath:<pathname>[=<options>]
                load native agent library by full pathname
-javaagent:<jarpath>[=<options>]
                load Java programming language agent, see java.lang.instrument
-splash:<imagepath>
                show splash screen with specified image
See http://www.oracle.com/technetwork/java/javase/documentation/index.html for more details.
C:\>
```



Descargar y configurar Eclipse

TUTORIALES

Español ▼

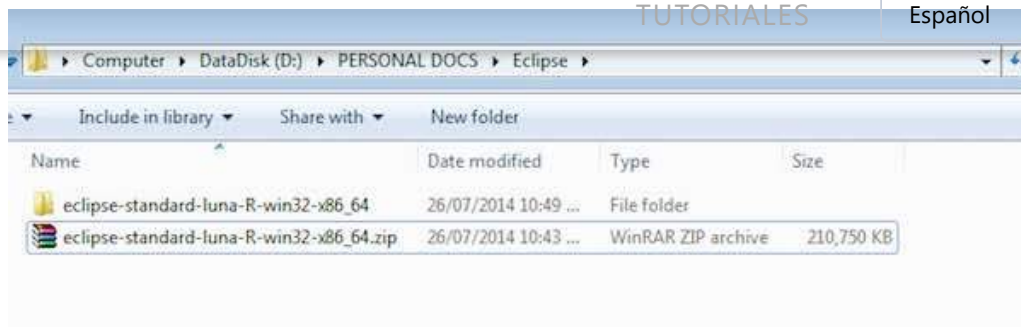
Paso 1: Vaya a la URL: <http://www.eclipse.org/downloads/> y descargue el archivo apropiado basado en la arquitectura de su sistema operativo.



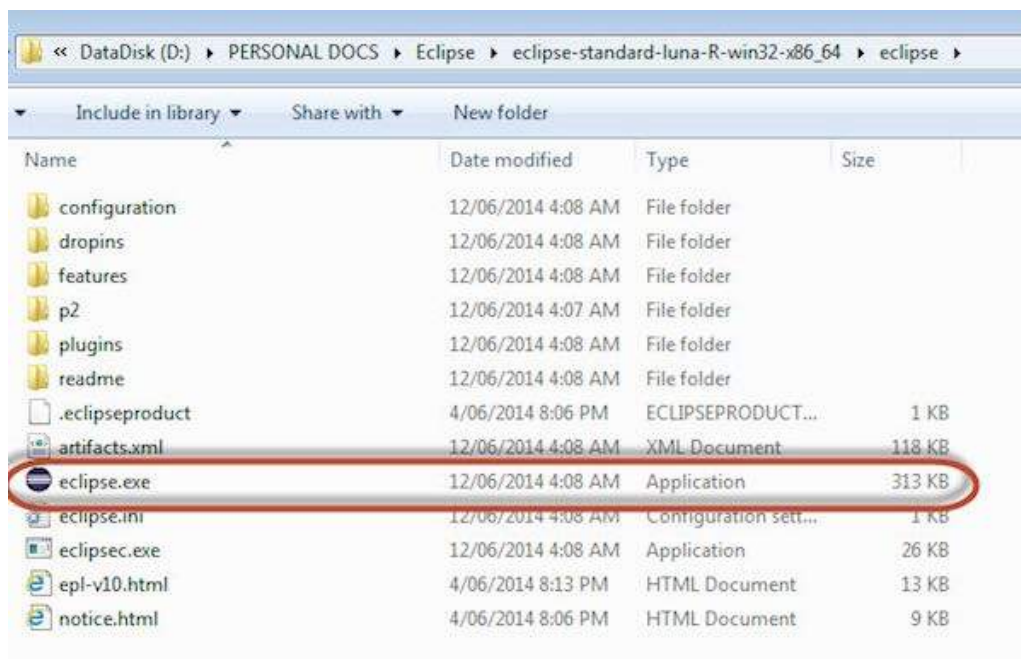
Paso 2: Haga clic en el botón 'Descargar'.



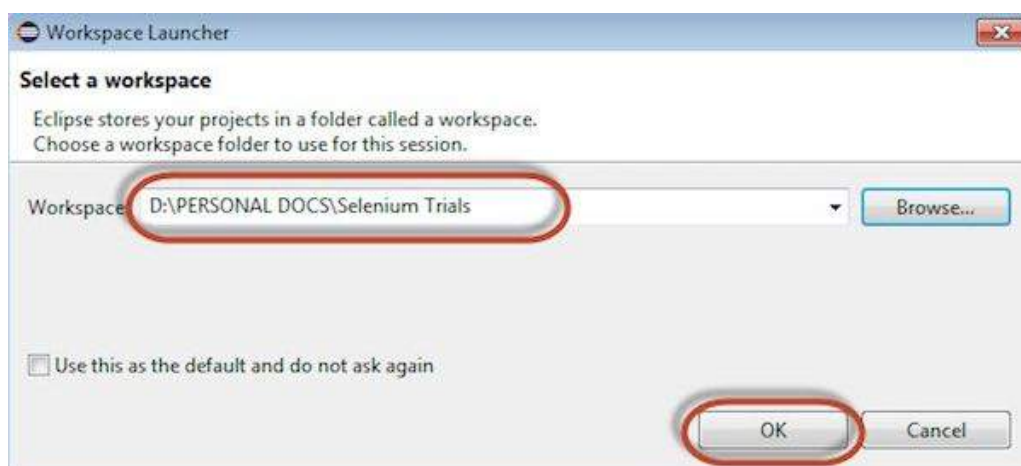
Paso 3: La descarga sería en un formato comprimido. Descomprimir el contenido.



Paso 4: Ubicar eclipse.exe y haga doble clic en el archivo.



Paso 5: Para configurar el espacio de trabajo, seleccione la ubicación en la que el desarrollo tiene que tener lugar.



Paso 6: La ventana de Eclipse abre como se muestra a continuación.



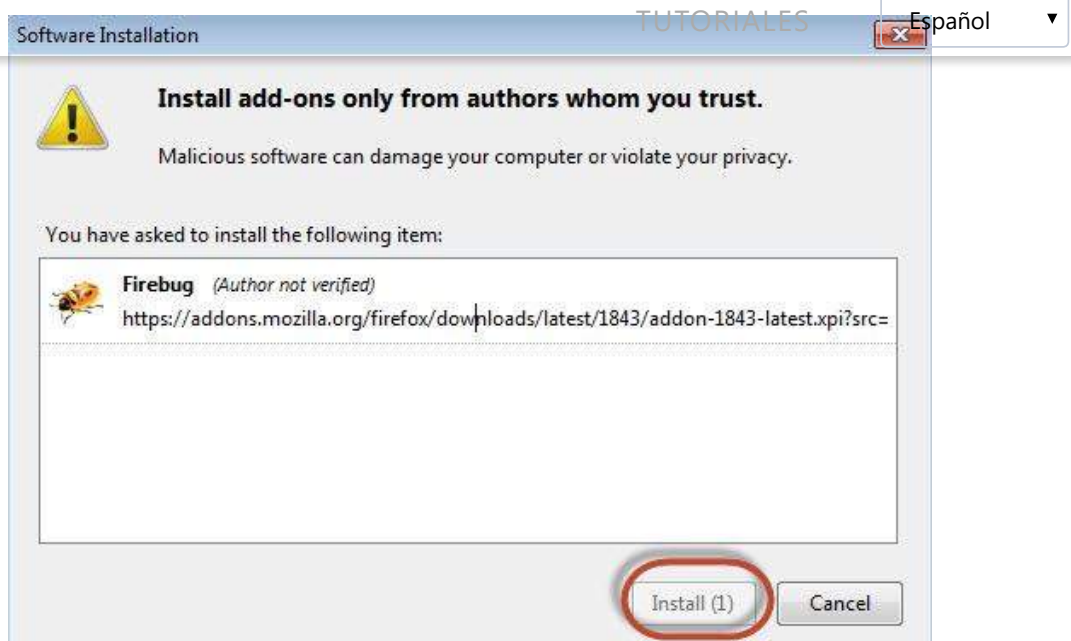
Configurar quemador, y la FirePath

Para trabajar con selenio RC o WebDriver, que necesitamos para localizar elementos en función de su XPath o identificación o nombre, etc. Con el fin de localizar un elemento, necesitamos herramientas / plugins.

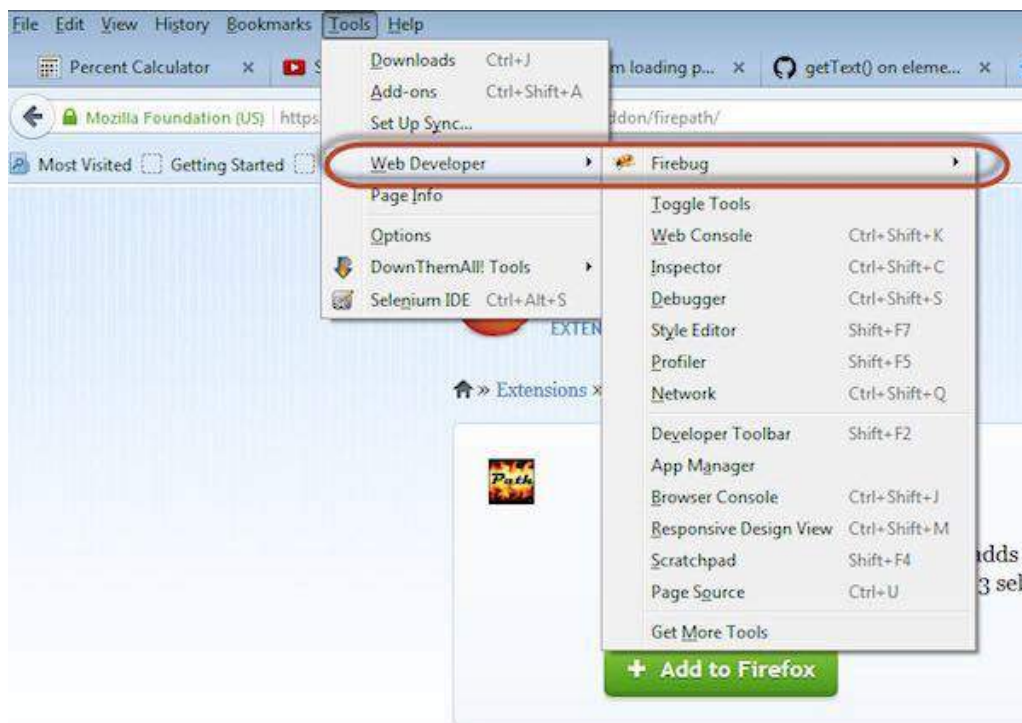
Paso 1: Vaya a la URL: <https://addons.mozilla.org/en-US/firefox/addon/firebug/> y descarga de plugins.



Paso 2: El instalador complemento se muestra al usuario y se instala cuando se hace clic en el botón "Instalar".



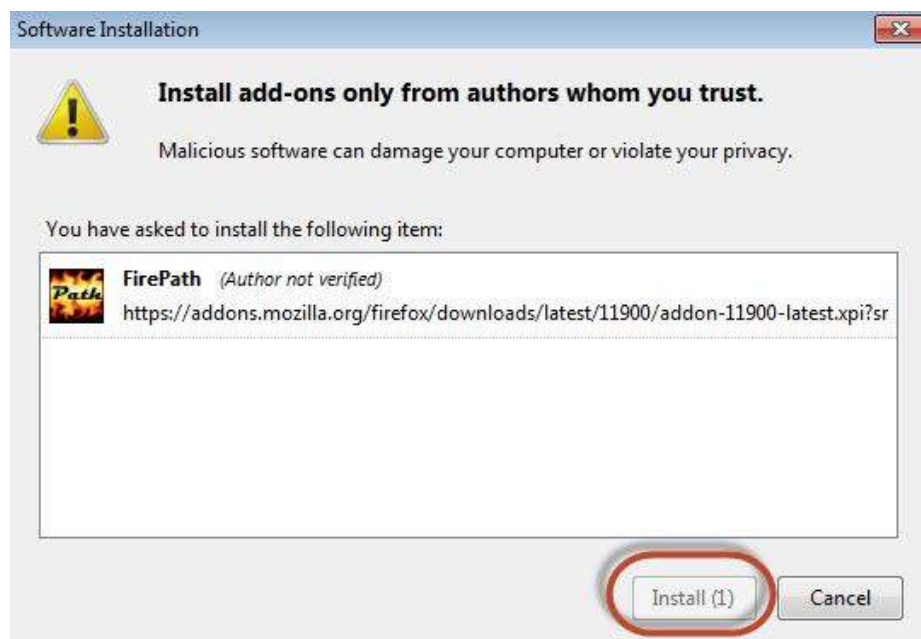
Paso 3: Una vez instalado, podemos lanzar el plugin navegando a "desarrollador web" >> "Firebug".



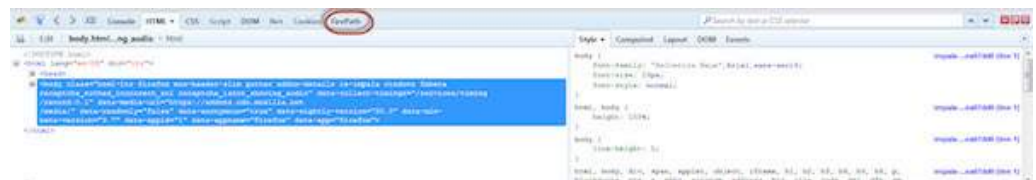
Paso 4: FirePath, un plugin que funciona dentro de Firebug, ayuda a los usuarios a agarrar el 'XPath' de un elemento. Instalar FirePath navegando a "https://addons.mozilla.org/en-US/firefox/addon/firepath/"



Paso 5: El instalador complemento se muestra al usuario y se instala cuando se hace clic en el botón "Instalar".



Paso 6: Ahora lanzar "Firebug" navegando a "Herramientas" >> "WebDeveloper" >> "Firebug".



Ejemplo

Ahora vamos a entender cómo usar quemador, y la FirePath con un ejemplo. Para la demostración, vamos a utilizar www.google.com y capturar las propiedades del cuadro de texto de "google.com".

Paso 1: En primer lugar, haga clic en el icono de la flecha como se destaca en la siguiente captura de pantalla y arrastrarlo al objeto para el cual nos gustaría capturar las propiedades. El HTML / DOM del objeto se



visualiza como se muestra a continuación. Somos capaces de captar el "ID" del cuadro de texto de entrada con la que podemos interactuar.

el "ID" del Español ▼



Paso 2: Para captar el XPath del objeto, vaya a la pestaña 'firepath' y realice los siguientes pasos.

- Haga clic en el icono de Spy.
- Seleccione el control para los que nos gustaría capturar el XPath
- XPath del control seleccionado se generaría.



Configurar Selenio RC

Ahora vamos a ver cómo configurar el control remoto selenio. Vamos a entender cómo desarrollar scripts con selenio RC en capítulos posteriores, sin embargo, por ahora, vamos a entender sólo la parte de configuración de la misma.

Paso 1: Vaya a la sección de descargas de selenio

<http://www.seleniumhq.org/download/> y descarga selenio servidor haciendo clic en su número de versión que se muestra a continuación.



TUTORIALES Español ▼

[edit this page](#)

[Projects](#)
[Download](#)
[Documentation](#)
[Support](#)
[About](#)

Selenium Downloads
[Latest Releases](#)
[Previous Releases](#)
[Source Code](#)
[Maven Information](#)

Downloads

Below is where you can find the latest releases of all the Selenium components. You can also find a list of [previous releases](#), [source code](#), and additional information for [Maven users](#) (Maven is a popular Java build tool).

Selenium IDE

Selenium IDE is a Firefox plugin which records and plays back user interactions with the browser. Use this to either create simple scripts or assist in exploratory testing. It can also export Remote Control or WebDriver scripts, though they tend to be somewhat brittle and should be overhauled into some sort of Page Object-y structure for any kind of resiliency.

Download latest released version **2.5.0** released on 01/Jan/2014 or view the [Release Notes](#) and then [install some plugins](#).

Selenium Server (formerly the Selenium RC Server)

The Selenium Server is needed in order to run either Selenium RC style scripts or Remote Selenium Webdriver ones. The 2.x server is a drop-in replacement for the old Selenium RC server and is [designed to be backwards compatible](#) with your existing infrastructure.

[Download version 2.42.2](#)

To use the Selenium Server in a Grid configuration [see the wiki page](#).

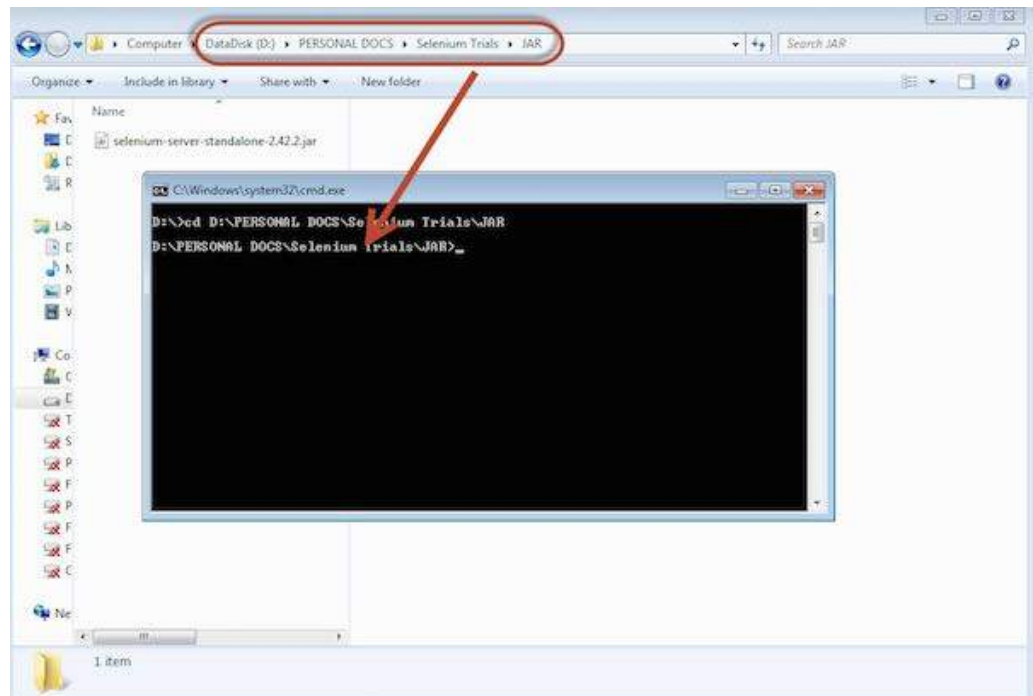
Donate to Selenium

with PayPal

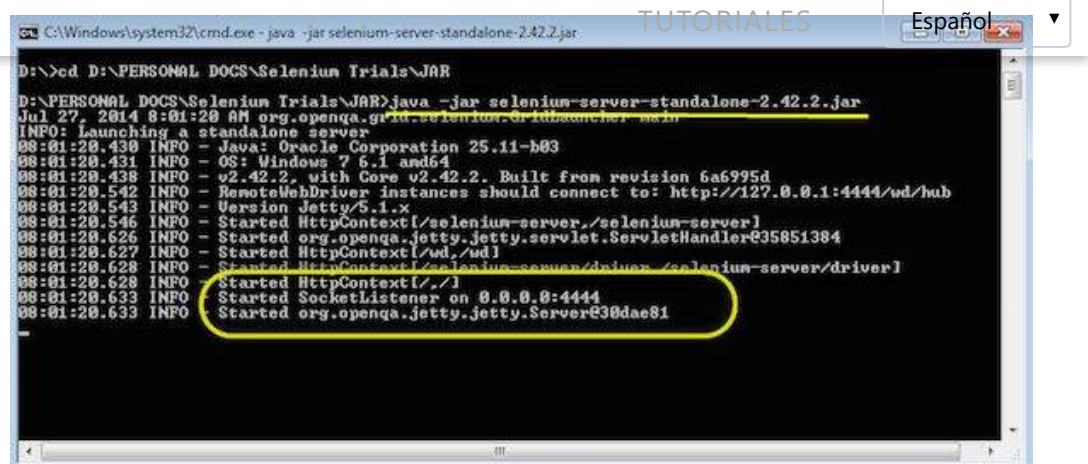
through sponsorship

You can [sponsor the Selenium project](#) if you'd like some public recognition of your generous contribution.

Paso 2: Después de la descarga, es necesario iniciar el Servidor de selenio. Para ello, símbolo del sistema abierto y vaya a la carpeta donde el archivo JAR descargado se mantiene como se muestra a continuación.



Paso 3: Para iniciar el servidor, utilice el jar 'comando java << >> descargado nombre del jar y si el JDK de Java está instalado correctamente, se llega a un mensaje de éxito, como se muestra a continuación. Ahora podemos empezar a escribir guiones Selenio RC.



El selenio configurar WebDriver

Ahora veamos cómo configurar WebDriver selenio. Vamos a entender cómo desarrollar scripts con selenio WebDriver en capítulos posteriores, sin embargo, por ahora, vamos a entender sólo la parte de configuración de la misma.

Paso 1: Vaya a la sección de descargas de selenio

<http://www.seleniumhq.org/download/> y descarga WebDriver selenio haciendo clic en su número de versión que se muestra a continuación.

The Internet Explorer Driver Server

This is required if you want to make use of the latest and greatest features of the WebDriver InternetExplorerDriver. Please make sure that this is available on your \$PATH (or %PATH% on Windows) in order for the IE Driver to work as expected.

Download version 2.42.0 for (recommended) [32 bit Windows IE](#) or [64 bit Windows IE](#)

[CHANGELOG](#)

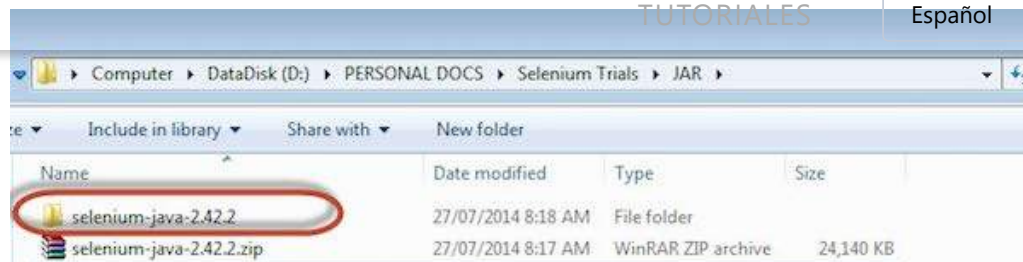
Selenium Client & WebDriver Language Bindings

In order to create scripts that interact with the Selenium Server (Selenium RC, Selenium Remote Webdriver) or create local Selenium WebDriver script you need to make use of language-specific client drivers. These languages include both 1.x and 2.x style clients.

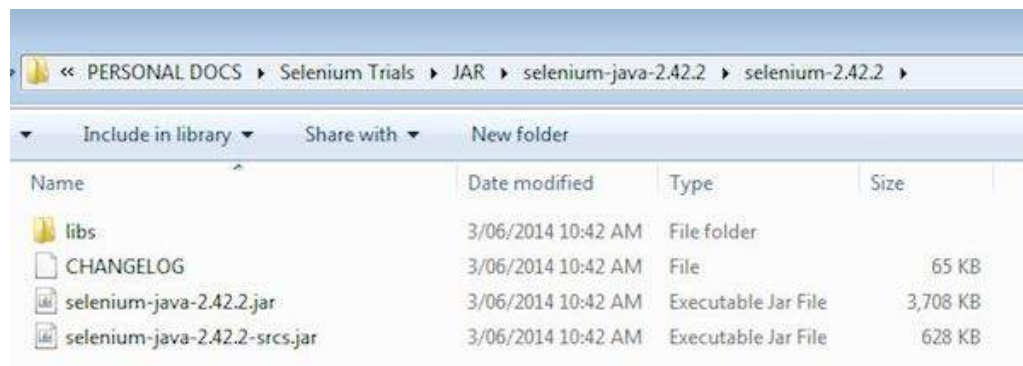
While language bindings for [other languages exist](#), these are the core ones that are supported by the main project hosted on google code.

Language	Client Version	Release Date	Download	Change log	Javadoc
Java	2.42.2	2014-06-03	Download	Change log	Javadoc
C#	2.42.0	2014-05-27	Download	Change log	API docs
Ruby	2.42.0	2014-05-22	Download	Change log	API docs
Python	2.42.1	2014-05-27	Download	Change log	API docs
Javascript (Node)	2.42.0	2014-05-22	Download	Change log	API docs

Paso 2: El archivo descargado está en formato comprimido y uno tiene que descomprimir el contenido de hacer un mapa de la carpeta del proyecto.



Paso 3: Se visualiza el contenido desabrochado como se muestra a continuación. ¿Cómo hacer un mapa de la carpeta del proyecto y cómo iniciar secuencias de comandos sería tratado en el capítulo WebDriver.



Selenio - RC

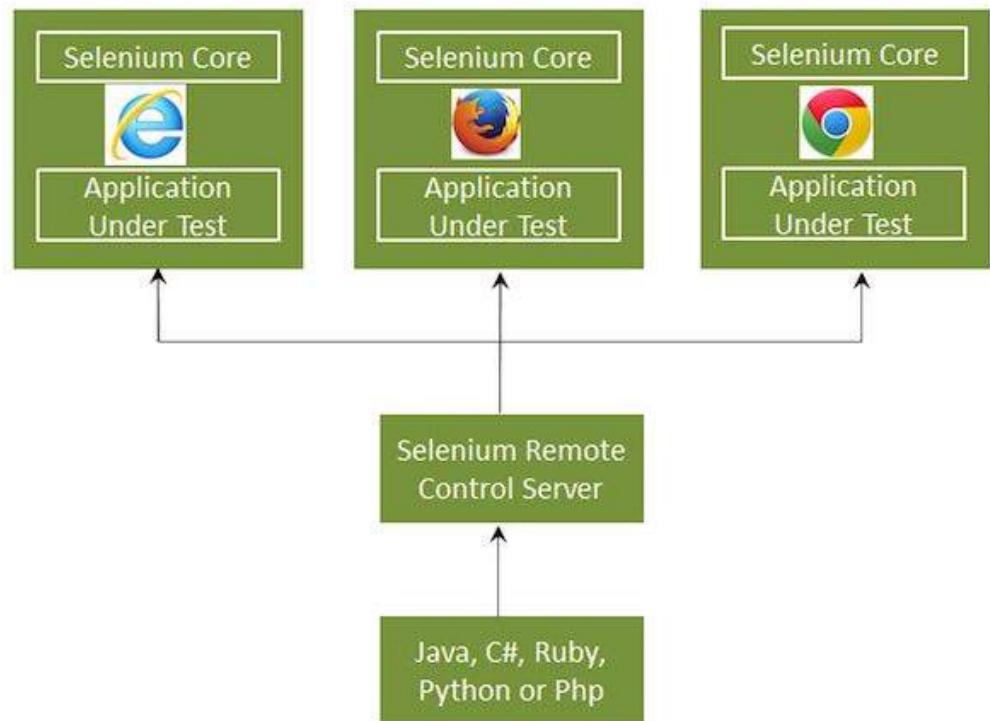
¿Cuál es Selenio RC?

Selenio control remoto (RC) fue el proyecto selenio principal que mantiene durante un largo tiempo antes de selenio WebDriver (Selenium 2,0) entró en existencia. Ahora Selenio RC es casi no está en uso, como WebDriver ofrece funciones más potentes, sin embargo los usuarios pueden seguir utilizando para desarrollar guiones RC.

Nos permite escribir pruebas automáticas de interfaz de usuario de aplicaciones web con la ayuda de toda la potencia de los lenguajes de programación como Java, C #, Perl, Python y PHP para crear pruebas más complejas, tales como archivos de lectura y escritura, se consulta una base de datos y enviar por correo electrónico los resultados de pruebas .

Arquitectura RC selenio

Selenium RC funciona de una manera tal que las bibliotecas de cliente puede comunicarse con el servidor de RC selenio pasando cada comando selenio para su ejecución. A continuación, el servidor pasa el comando selenio para el navegador mediante los comandos de selenio-Core JavaScript.



El selenium RC viene en dos partes.

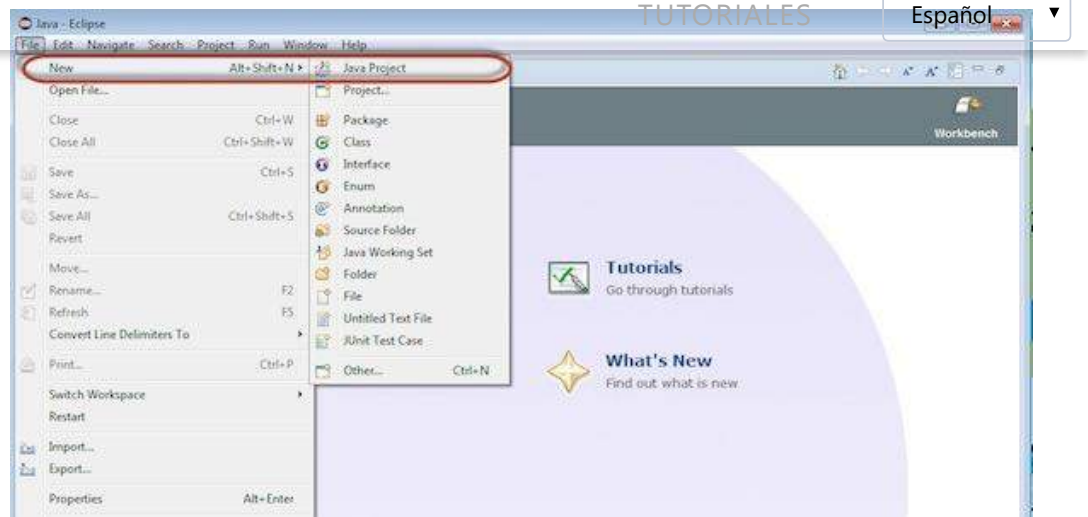
- El servidor de selenium lanza y mata a los navegadores. Además de eso, se interpreta y ejecuta los comandos Selenese. También actúa como un proxy HTTP mediante la interceptación y la verificación de los mensajes HTTP transmitidos entre el navegador y la aplicación que se está probando.
- bibliotecas de cliente que proporcionan una interfaz entre cada uno de los lenguajes de programación (Java, C #, Perl, Python y PHP) y el selenium RC-servidor.

RC Scripting

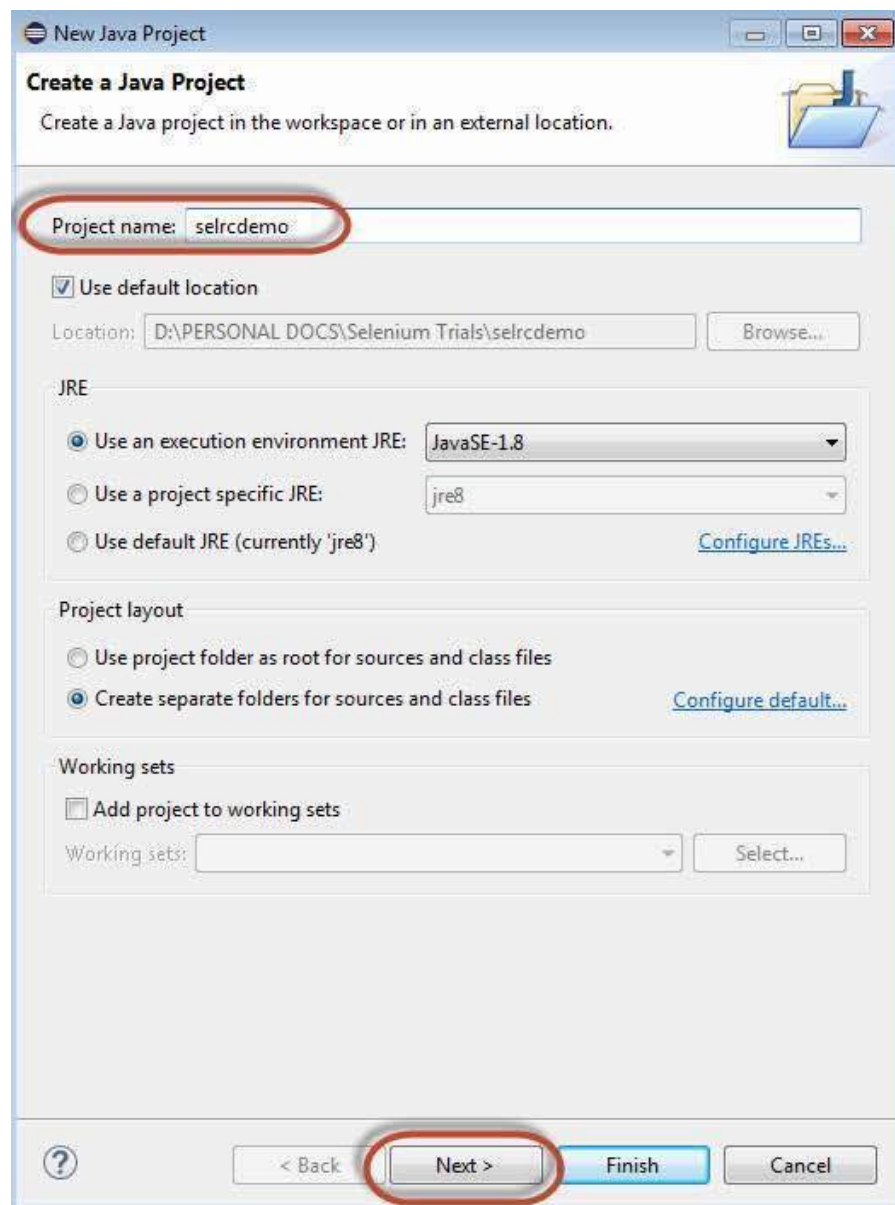
Ahora vamos a escribir un script de ejemplo utilizando selenium control remoto. Usemos <http://www.calculator.net/> para la comprensión de selenium RC. Vamos a realizar un cálculo de porcentaje de uso de 'Percent Calculator' que está presente bajo el módulo de 'Matemáticas calculadoras'.

Paso 1: Control remoto de inicio de selenium (con la ayuda de línea de comandos).

Paso 2: Después de lanzar Selenium RC, Eclipse abierto y crear un "Nuevo proyecto", como se muestra a continuación.



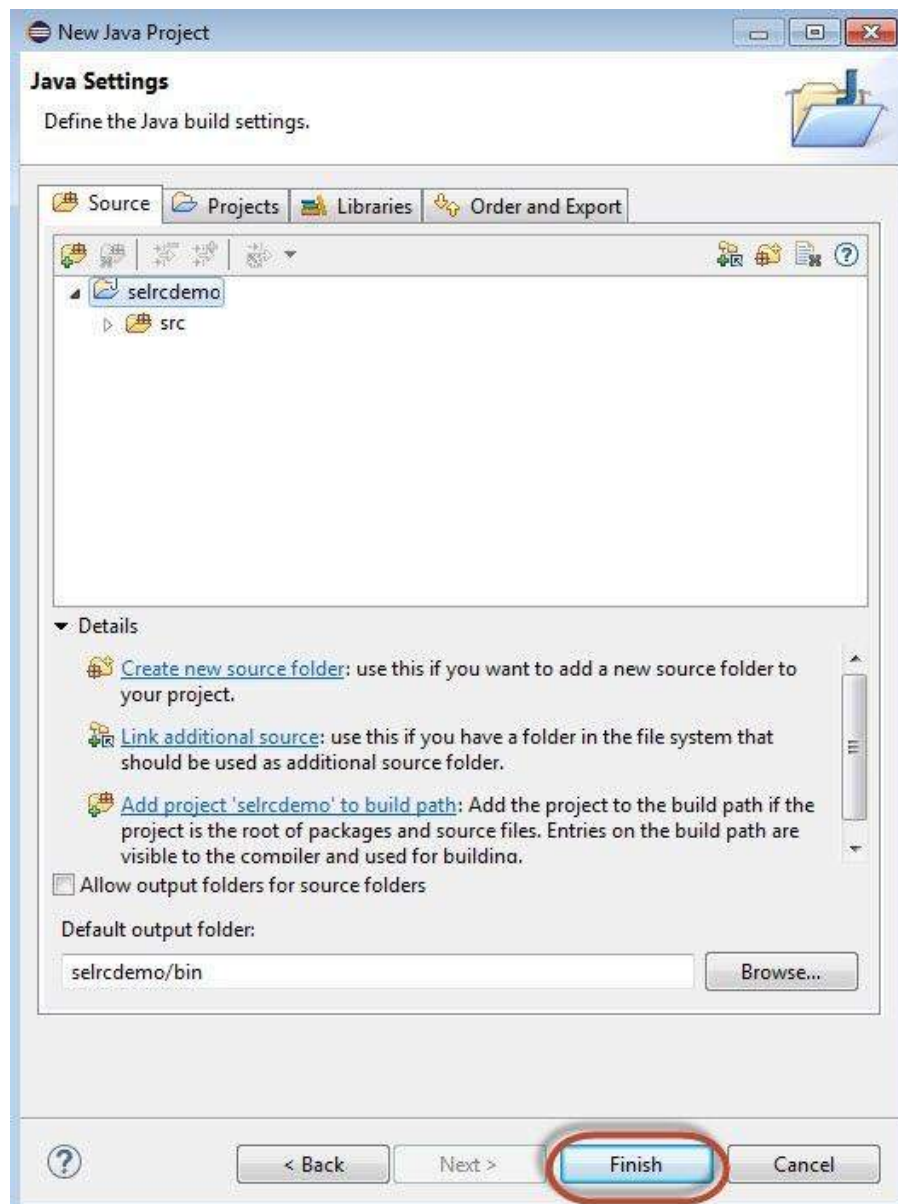
Paso 3: Introduzca el nombre del proyecto y haga clic en el botón 'Siguiente'.

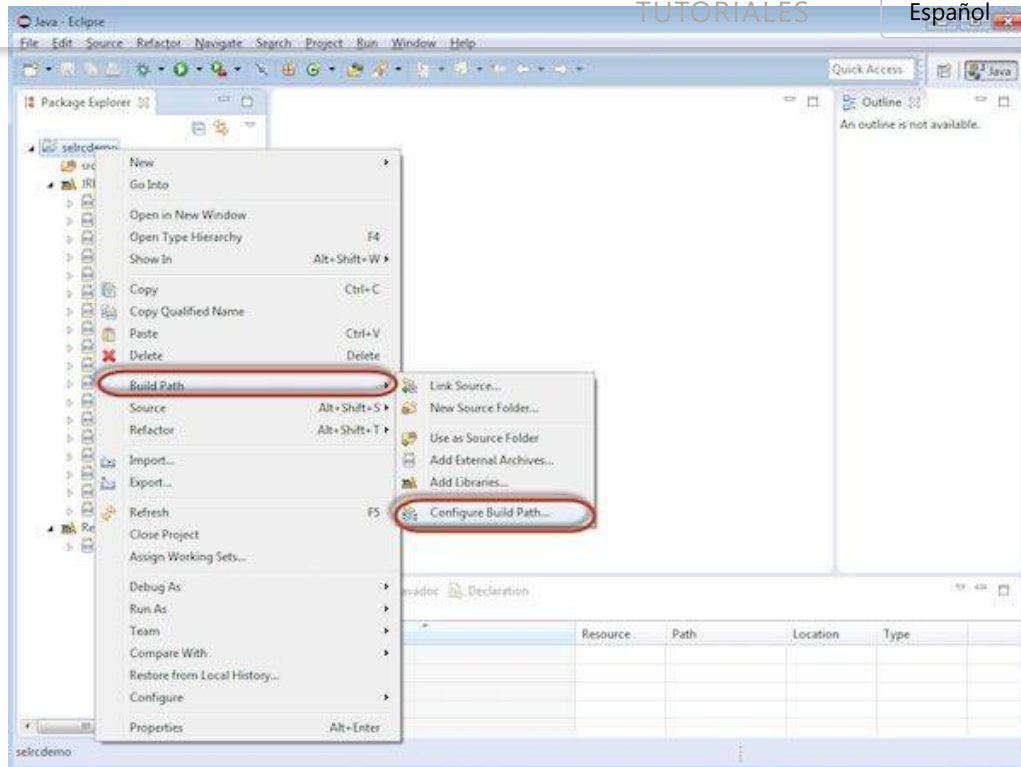


Paso 4: Verificar la Fuente, Proyectos, Bibliotecas, y la carpeta de salida, y haga clic en "Finalizar".

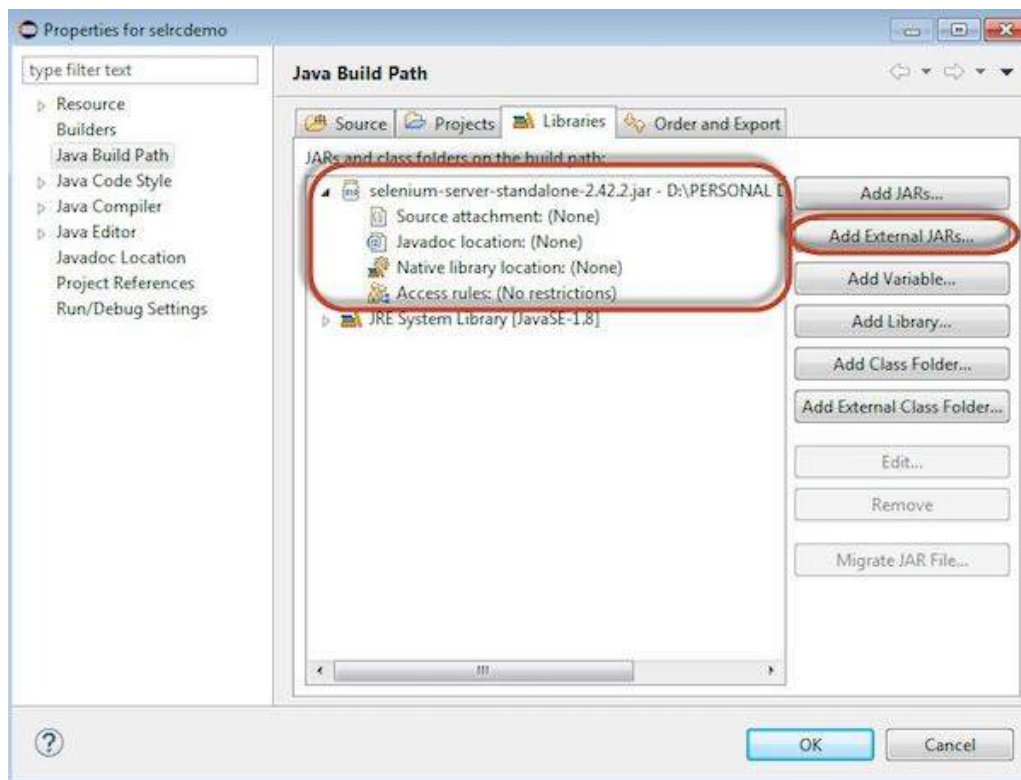
TUTORIALES

Español

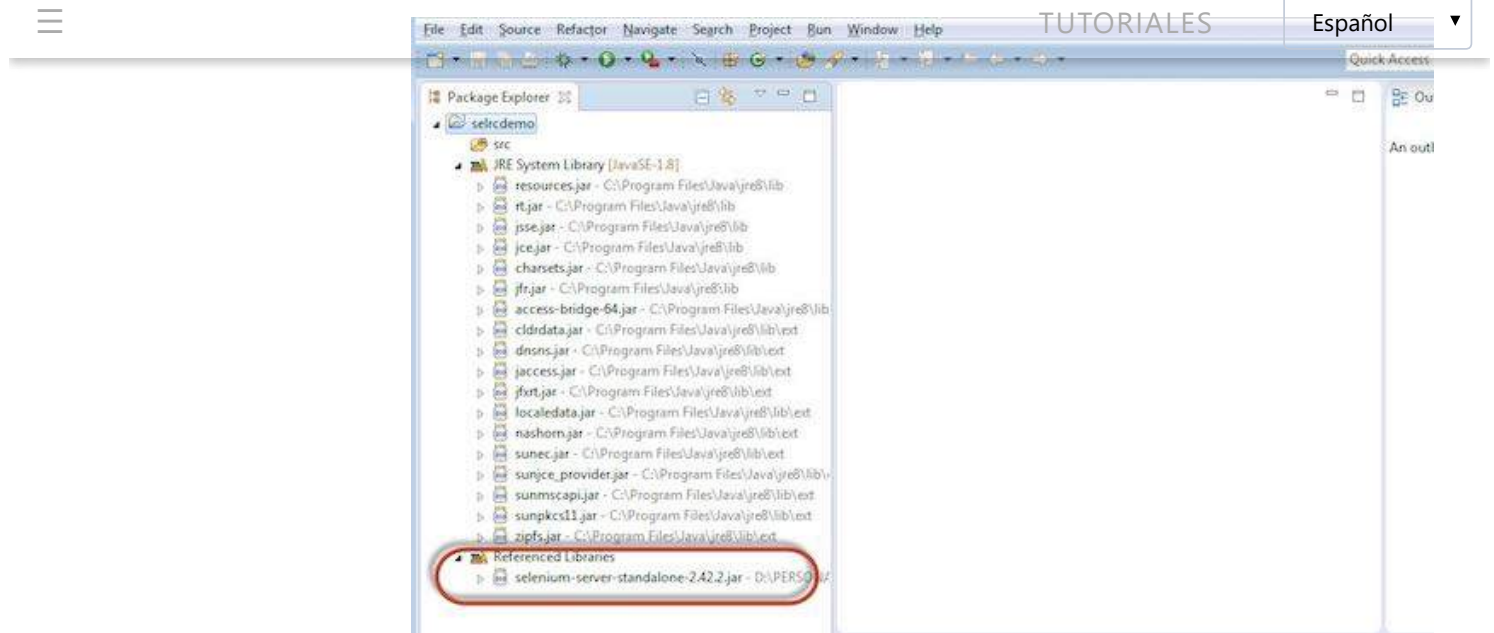
**Paso 5: Haga clic derecho en el envase del "proyecto" y seleccione 'Configurar Vía de construcción'.**



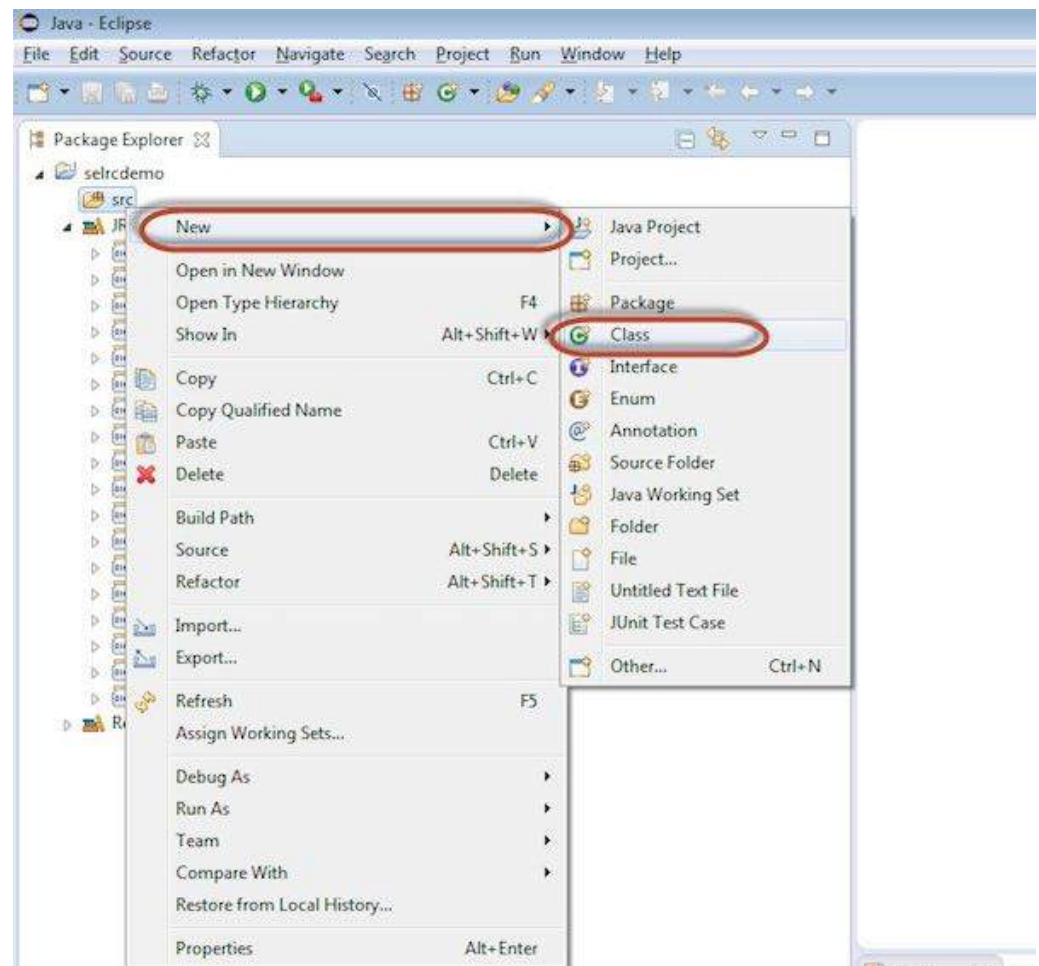
Paso 6: Propiedades de 'selrcdemo' se abre. Navegue a la pestaña 'Bibliotecas' y selecciona 'Añadir JAR externos'. Elija el archivo jar Selenium RC que hemos descargado y que aparecería como se muestra a continuación.



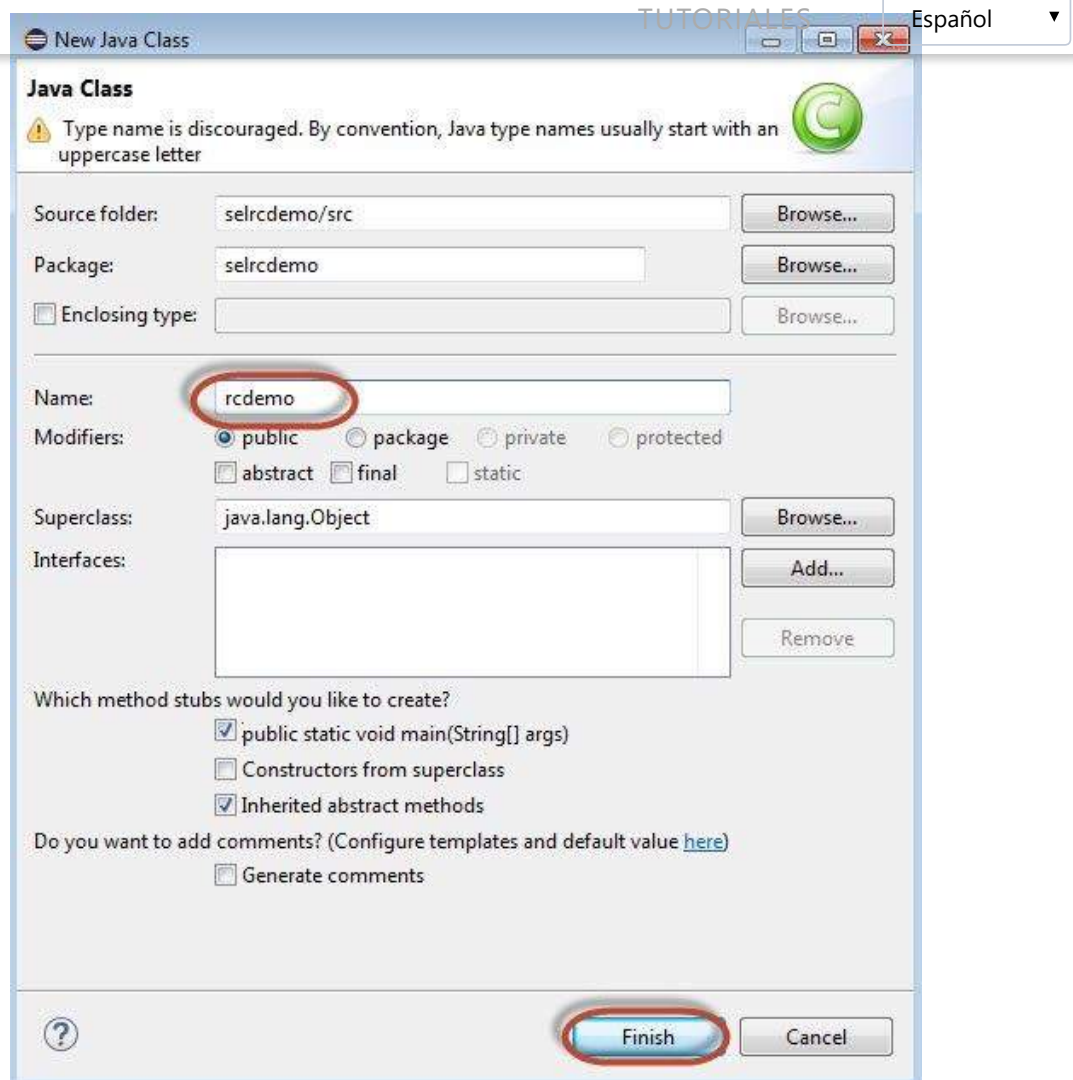
Paso 7: Las bibliotecas de referencia se muestran como se muestra a continuación.



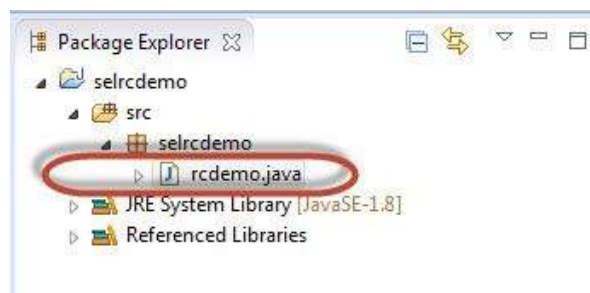
Paso 8: Crear un nuevo archivo de clase mediante la realización de un clic derecho sobre la carpeta 'src' y seleccionar "Nuevo" >> "clase".



Paso 9: Introduzca el nombre del archivo de clase y permitir 'public void' como se muestra a continuación.



Paso 10: La Creado Class se crea bajo la estructura de carpetas como se muestra a continuación.



Paso 11: Ahora es el momento para la codificación. El siguiente código tiene comentarios que hay en él para hacer los lectores a entender lo que se ha planteado.

```
package selrcdemo;
import com.thoughtworks.selenium.DefaultSelenium;
import com.thoughtworks.selenium.Selenium;
public class rcdemo
{
    public static void main(String[] args) throws InterruptedException
    {
        // Instatiate the RC Server
        Selenium selenium = new DefaultSelenium("localhost", 4444 , "firefox"
        selenium.start(); // Start
        selenium.open("/"); // Open the URL
        selenium.windowMaximize();

        // Click on Link Math Calculator
        selenium.click("xpath=//*[@id='menu']/div[3]/a");
        Thread.sleep(2500); // Wait for page load

        // Click on Link Percent Calculator
        selenium.click("xpath=//*[@id='menu']/div[4]/div[3]/a");
        Thread.sleep(4000); // Wait for page load

        // Focus on text Box
        selenium.focus("name=cpar1");
        // enter a value in Text box 1
        selenium.type("css=input[name=\"cpar1\"]", "10");

        // enter a value in Text box 2
        selenium.focus("name=cpar2");
        selenium.type("css=input[name=\"cpar2\"]", "50");

        // Click Calculate button
        selenium.click("xpath=//*[@id='content']/table/tbody/tr/td[2]/input");

        // verify if the result is 5
        String result = selenium.getText("//*[id='content']/p[2]");

        if (result == "5")
        {
            System.out.println("Pass");
        }
        else
        {
            System.out.println("Fail");
        }
    }
}
```

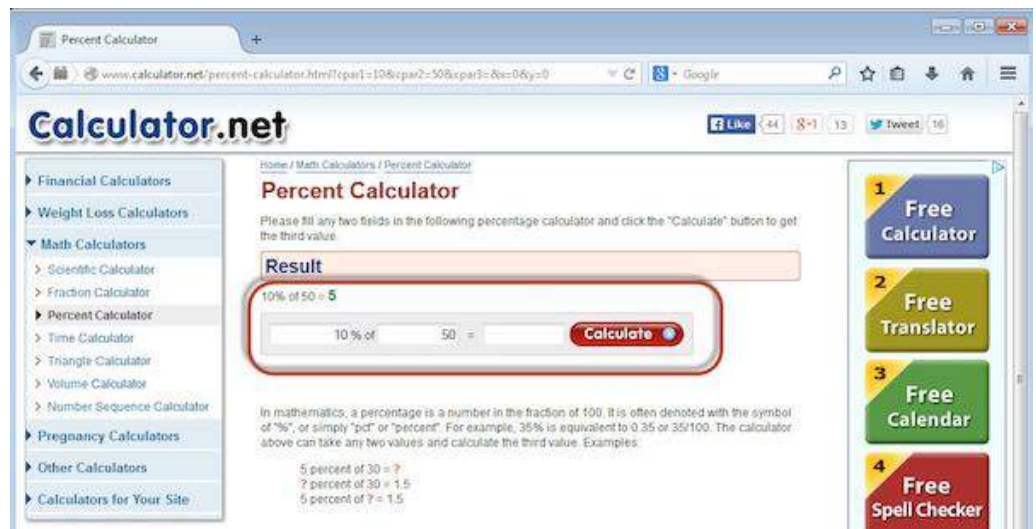
Paso 11: Ahora, vamos a ejecutar la secuencia de comandos haciendo clic en botón "Ejecutar".



Paso 12: La secuencia de comandos se comienza la ejecución y el usuario sería capaz de ver el historial de comandos en la pestaña 'Comando Historia'.



Paso 13: El estado final de la aplicación se muestra a continuación. El porcentaje se calcula y se muestra el resultado en la pantalla como se muestra a continuación.



Paso 14: La salida de la prueba se imprime en la consola de Eclipse como se muestra a continuación como se ha impreso la salida de la consola. En tiempo real el resultado se escribe en un archivo HTML o en un archivo de texto simple.



Selenium - Comandos Selenese

Selenium - Comandos Selenese

Un comando se refiere a lo que el selenium tiene que hacer y comandos en el selenium son de tres tipos. Haga clic en cada uno de ellos para saber más acerca de los comandos.

- **[Comportamiento \(selenium_commands_actions.html\)](#)**
- **[Los descriptores de acceso \(selenium_commands_accessors.html\)](#)**
- **[Las afirmaciones \(selenium_commands_assertions.html\)](#)**

localizadores

Centradores del elemento ayudan selenium para identificar el elemento HTML el comando se refiere a. Todos estos localizadores pueden ser identificados con la ayuda de FirePath y el plugin de Mozilla Firebug. Por favor, consulte el capítulo Configuración del entorno para obtener más detalles.

- **identificador = id** Seleccione el elemento con el atributo "id" especificada y si no hay ninguna coincidencia, seleccionar el primer elemento cuyo atributo es la Identificación del @Name.
- **id = ID** Seleccione el elemento con el atributo "id" especificado.
- **name = nombre** Seleccione el primer elemento con el atributo "nombre" especifica
- **dom = javascriptExpression** El selenium se encuentra un elemento mediante la evaluación de la cadena especificada que nos permite recorrer a través del modelo de objetos de documento HTML usando **JavaScript**. Los usuarios no pueden devolver un valor, pero pueden evaluar como una expresión en el bloque.
- **XPath = XPathExpression** Localizar un elemento utilizando una expresión XPath.
- **link = textpattern** Seleccione el elemento de enlace (dentro de las etiquetas de anclaje), que contiene el texto que coincida con el patrón especificado.
- **css = cssSelectorSyntax** Seleccione el elemento mediante el selector CSS.

Selenium - WebDriver

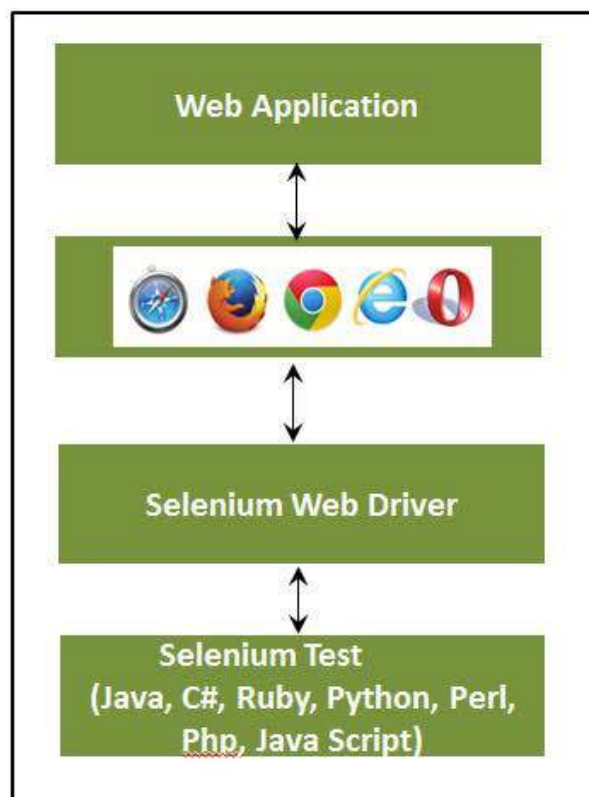
El selenium WebDriver

WebDriver es una herramienta para automatizar las pruebas de aplicaciones web. Es conocido popularmente como el selenio 2.0. WebDriver utiliza un marco subyacente diferente, mientras que el selenio RC utiliza JavaScript Selenio-Core incrustado dentro del navegador, que tiene algunas limitaciones. WebDriver interactúa directamente con el navegador sin ningún intermediario, a diferencia de selenio RC que depende de un servidor. Se utiliza en el contexto siguiente:

- las pruebas multi-navegador que incluye una funcionalidad mejorada para los navegadores que no está bien soportado por Selenio RC (Selenio 1.0).
- Manejo de múltiples tramas, múltiples ventanas del navegador, ventanas emergentes y alertas.
- la navegación de páginas complejas.
- de usuario de navegación avanzada, tales como arrastrar y soltar.
- elementos de la interfaz basada en AJAX.

Arquitectura

WebDriver se explica mejor con un simple diagrama de la arquitectura, como se muestra a continuación.



El selenio RC Vs WebDriver

El selenio RC	WebDriver selenio
---------------	-------------------

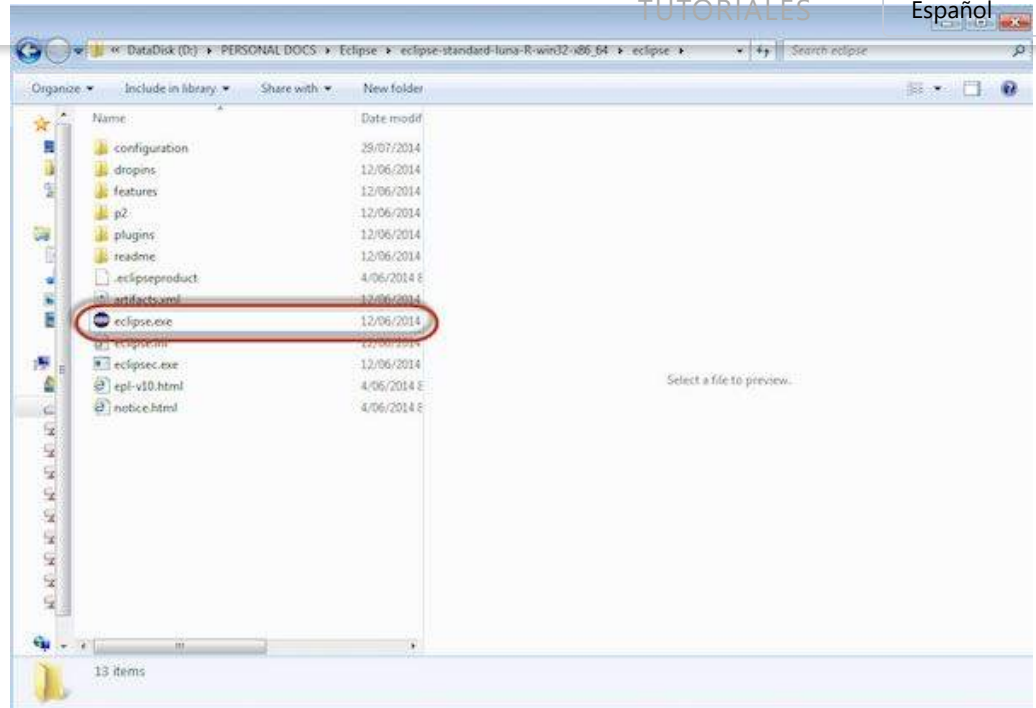


La arquitectura de selenio RC es complicada, ya que el servidor tiene que estar en funcionamiento antes de comenzar una prueba.	La arquitectura de WebDriver es más simple que el selenio RC, ya que controla el explorador en el nivel de sistema operativo.
servidor de selenio actúa como un intermediario entre el navegador y los comandos Selenese.	WebDriver interactúa directamente con el navegador y utiliza el motor del navegador para controlarlo.
El selenio ejecución del script RC es más lento, ya que utiliza una Javascript para interactuar con RC.	WebDriver es más rápido, ya que interactúa directamente con el navegador.
RC selenio no puede soportar la ejecución sin cabeza, ya que necesita un verdadero navegador para trabajar con ellos.	WebDriver puede apoyar la ejecución sin cabeza.
Es una API simple y pequeño.	Complex y una gran API poco en comparación con RC.
Menos API orientada a objetos.	Puramente objeto de la API orientada.
No se puede probar las aplicaciones móviles.	Puede probar las aplicaciones iPhone / Android.

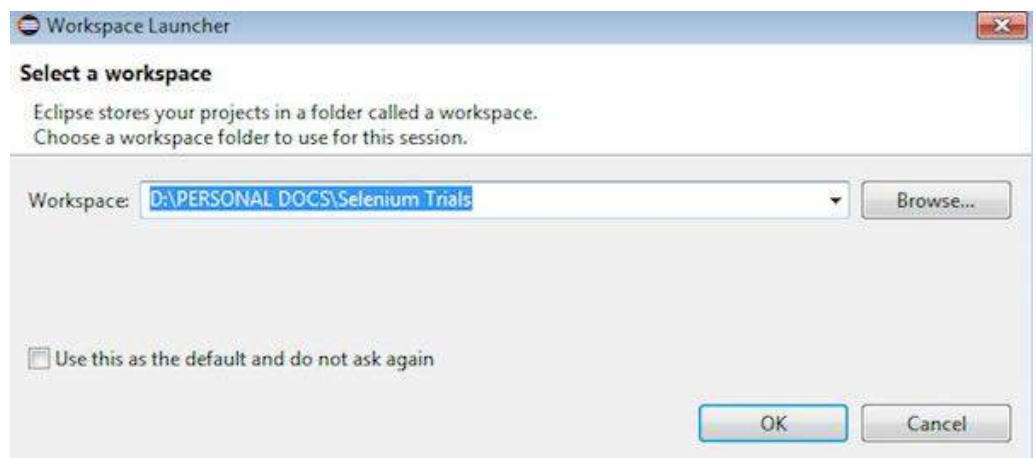
Secuencias de comandos usando WebDriver

Vamos a entender cómo trabajar con WebDriver. Para la demostración, usaremos <http://www.calculator.net/>. Vamos a realizar una "calculadora ciento" que se encuentra en "Calculadora de Matemáticas". Ya hemos descargado el archivo JAR WebDriver requerido de. Consulte el capítulo "Configuración del Medio Ambiente" para más detalles.

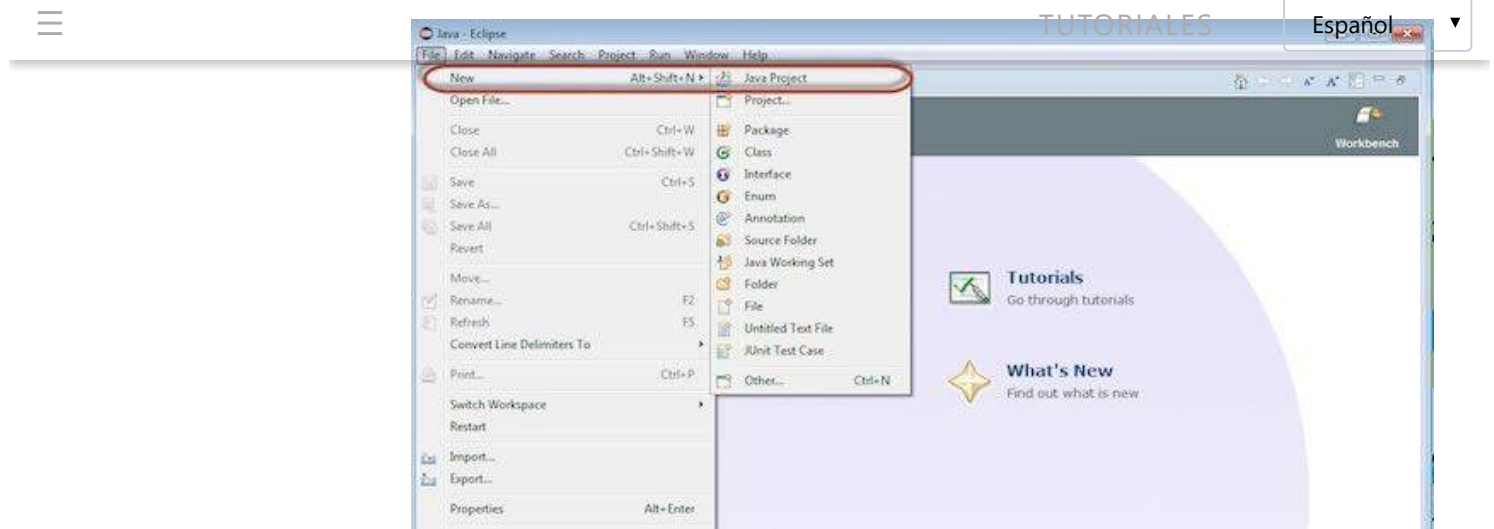
Paso 1: Poner en marcha "Eclipse" de la carpeta extraída Eclipse.



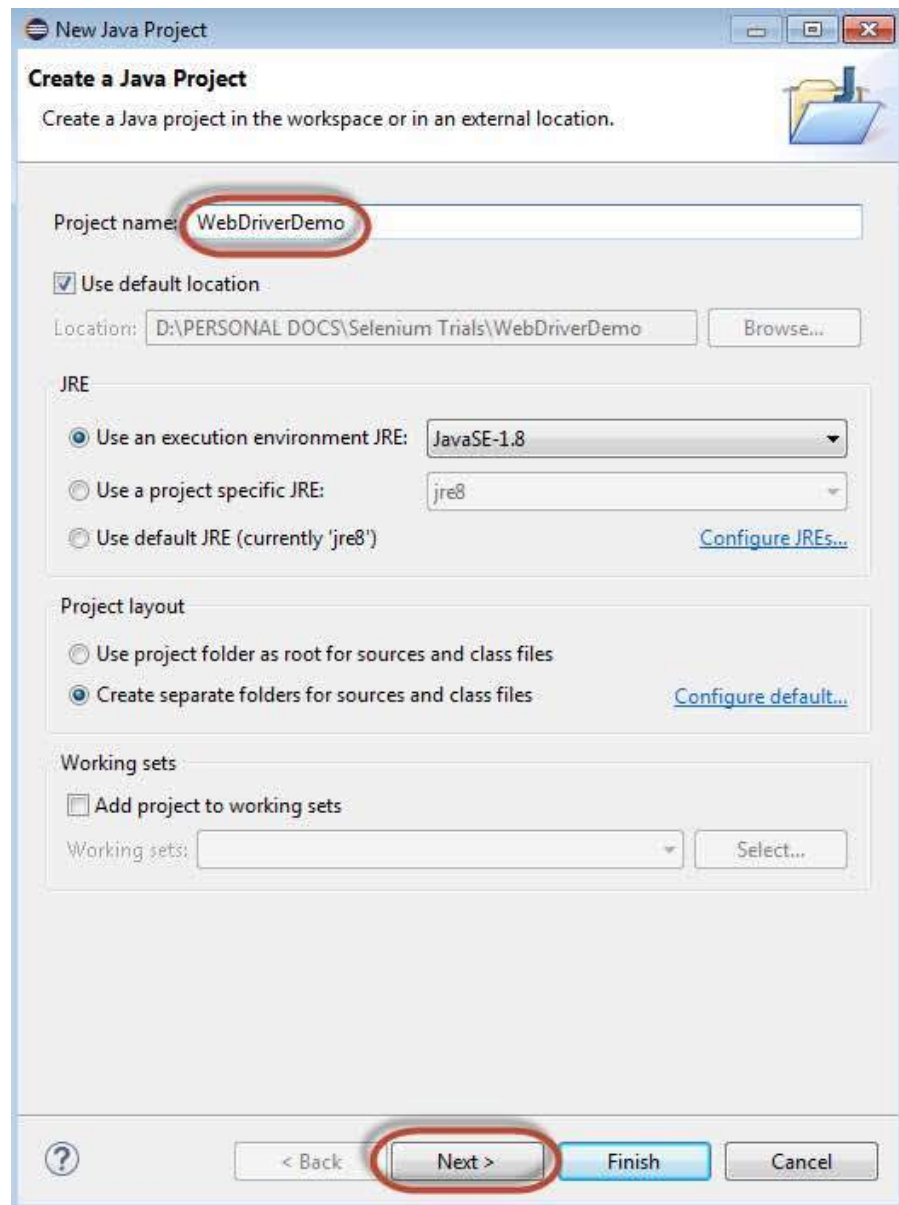
Paso 2: Seleccionar el espacio de trabajo haciendo clic en el botón "Examinar".



Paso 3: Ahora crea un "nuevo proyecto" en el menú "Archivo".

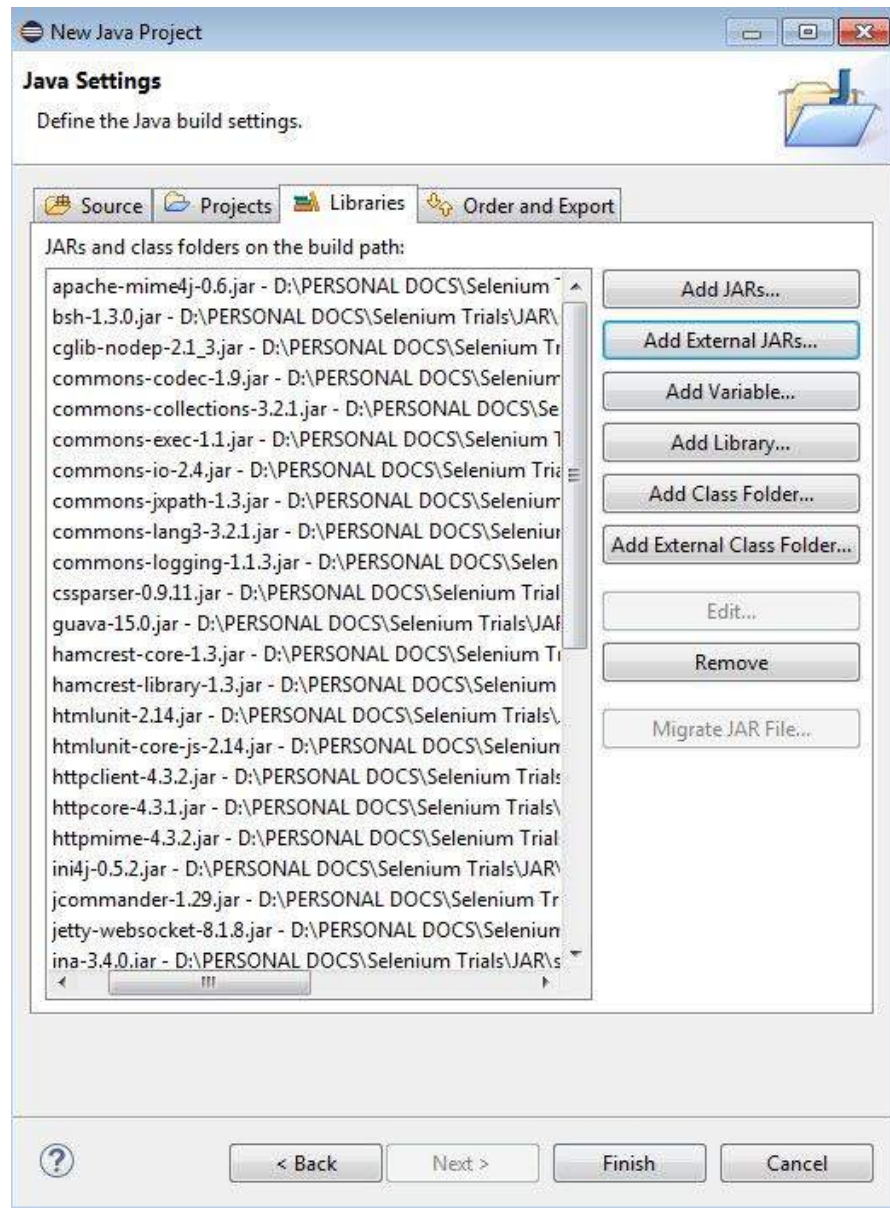


Paso 4: Introduzca el nombre del proyecto y haga clic en "Siguiete".

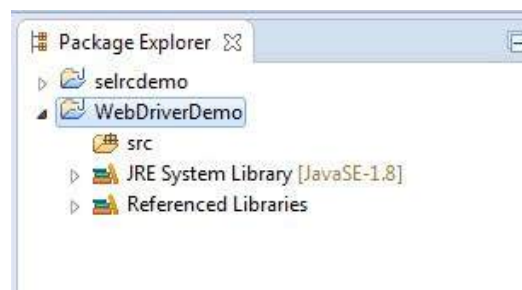


Paso 5: Ir a la pestaña Bibliotecas y seleccionar todo el que hemos descargado JAR. Añadir referencia a todos los del tarro de carpeta de selenium

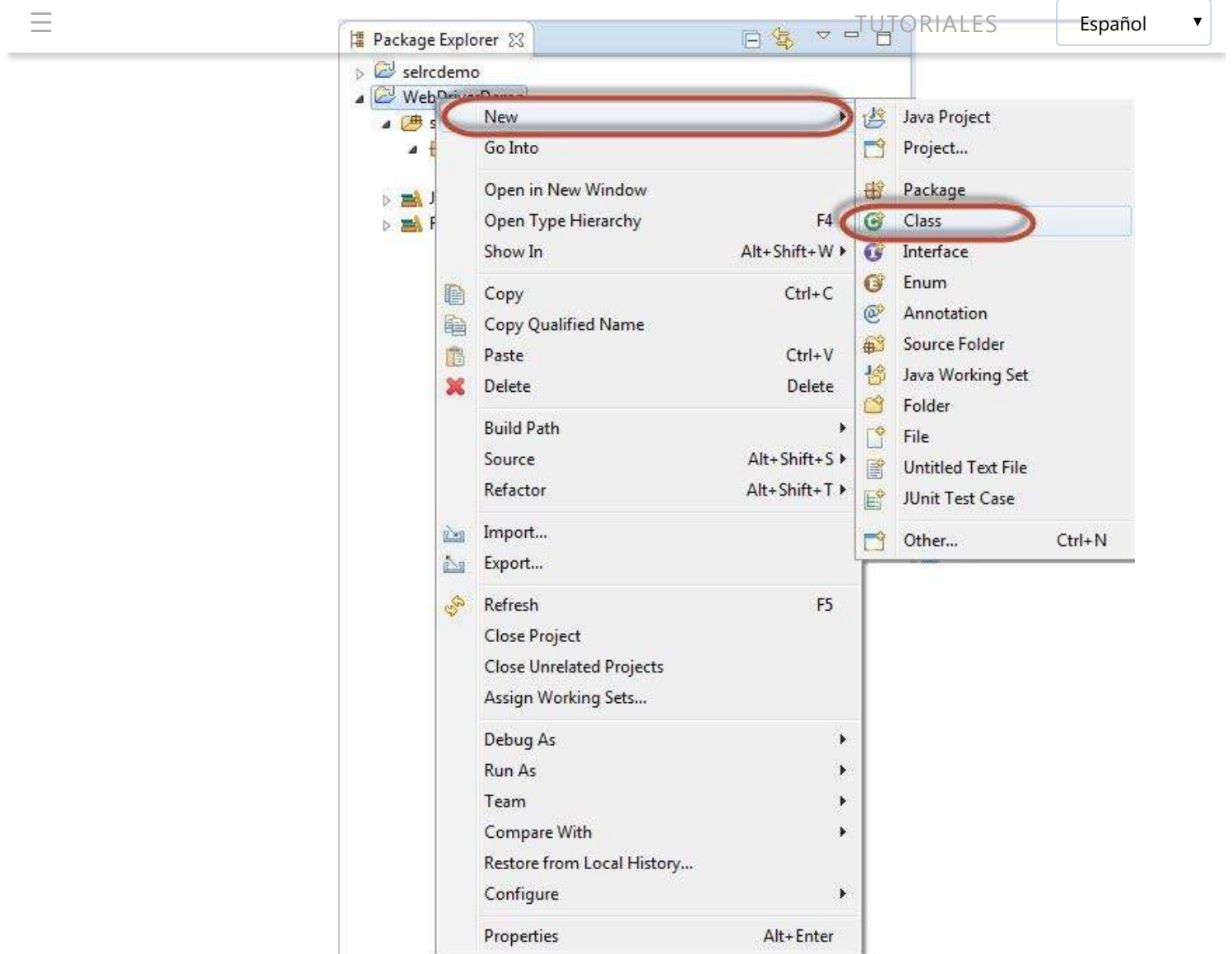
WebDriver biblioteca y también el selenium-java-2.42.2.jar y selenium-java-2.42.2-srscs.jar.



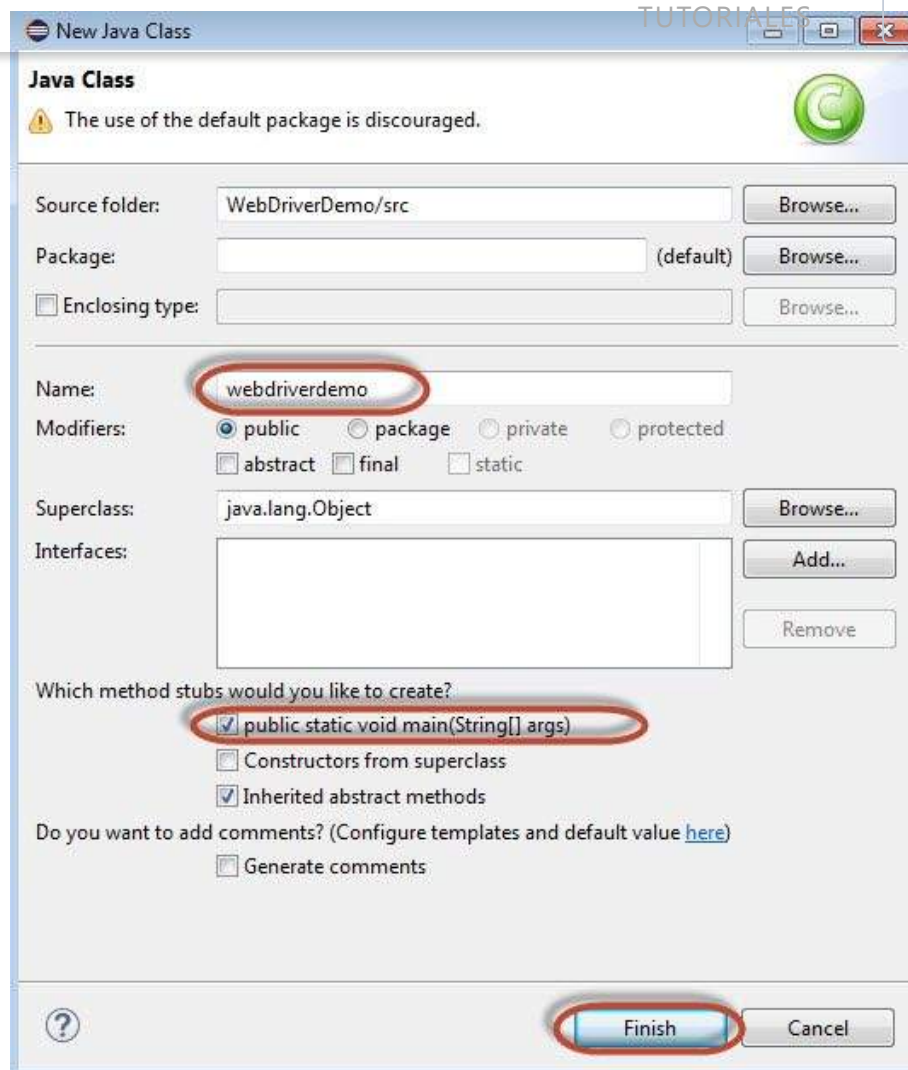
Paso 6: Se crea el paquete como se muestra a continuación.



Paso 7: Ahora haga clic derecho en el paquete y seleccionar "Nuevo" >> "clase" para crear una "clase".



Paso 8: Ahora el nombre de la clase y hacer que la función principal.



Paso 9: El esquema de la clase se muestra a continuación.



Paso 10: Ahora es el momento de codificar. La siguiente secuencia de comandos es más fácil de entender, como lo ha hecho comentarios que hay en ella para explicar los pasos claramente. Por favor, eche un vistazo a los "localizadores" capítulo para comprender cómo capturar las propiedades del objeto.

```
import java.util.concurrent.TimeUnit;
import org.openqa.selenium.*;
import org.openqa.selenium.firefox.FirefoxDriver;
public class webdriverdemo
{
    public static void main(String[] args)
    {
        WebDriver driver = new FirefoxDriver();
        //Puts an Implicit wait, Will wait for 10 seconds before throwing ex
        driver.manage().timeouts().implicitlyWait(10, TimeUnit.SECONDS);

        //Launch website
        driver.navigate().to("http://www.calculator.net/");

        //Maximize the browser
        driver.manage().window().maximize();

        // Click on Math Calculators
        driver.findElement(By.xpath(".*[@id='menu']/div[3]/a")).click();

        // Click on Percent Calculators
        driver.findElement(By.xpath(".*[@id='menu']/div[4]/div[3]/a")).cli

        // Enter value 10 in the first number of the percent Calculator
        driver.findElement(By.id("cpar1")).sendKeys("10");

        // Enter value 50 in the second number of the percent Calculator
        driver.findElement(By.id("cpar2")).sendKeys("50");

        // Click Calculate Button
        driver.findElement(By.xpath(".*[@id='content']/table/tbody/tr[2]/t

        // Get the Result Text based on its xpath
        String result = driver.findElement(By.xpath(".*[@id='content']/p[2

        // Print a Log In message to the screen
        System.out.println(" The Result is " + result);

        //Close the Browser.
        driver.close();
    }
}
```

Paso 11: La salida del script anterior se imprimiría en la consola.





La siguiente tabla muestra algunos de los comandos más utilizados en WebDriver junto con su sintaxis.

Mando	Descripción
driver.get ("URL")	Para navegar a una aplicación.
element.sendKeys ("inputText")	Introduzca un texto en un cuadro de entrada.
element.clear ()	Borrar el contenido de la caja de entrada.
select.deselectAll ()	Deseleccionar todas las opciones desde la primera vez que seleccione en la página.
select.selectByVisibleText ("texto")	Selecione la opción con la entrada especificada por el usuario.
driver.switchTo (). ventana ("windowName")	Mover el foco de una ventana a otra.
driver.switchTo (). marco ("frameName")	Oscilación de cuadro a cuadro.
driver.switchTo (). alerta ()	Ayuda en el manejo de alertas.
driver.navigate (). a ("URL")	Navegue a la URL.
driver.navigate (). forward ()	Para navegar hacia delante.
driver.navigate (). atrás ()	Para navegar hacia atrás.
driver.close ()	Cierra el navegador actual asociado con el conductor.
driver.quit ()	Sale del conductor y cierra toda la ventana asociada de dicho conductor.
driver.refresh ()	Actualiza la página actual.

Selenium - localizadores

Selenium - localizadores

los elementos de colocación en selenium WebDriver se lleva a cabo con la ayuda de findElement () y findElements () métodos proporcionados por WebDriver y la clase WebElement.



- `findElement ()` devuelve un objeto `WebElement` basado en un criterio de búsqueda especificados o termina por lanzar una excepción si no encuentra ningún elemento que cumpla con los criterios de búsqueda.
- `findElements ()` devuelve una lista de `WebElements` que coinciden con los criterios de búsqueda. Si no se encuentran elementos, devuelve una lista vacía.

La siguiente tabla muestra todas las sintaxis de Java para los elementos de selenio WebDriver localizar.

Método	Sintaxis	Descripción
por ID	<code>driver.findElement (By.id (<ID de elemento>))</code>	Localiza un elemento con el atributo ID
Por nombre	<code>driver.findElement (By.name (<nombre del elemento>))</code>	Localiza un elemento con el atributo Nombre
Por nombre de la clase	<code>driver.findElement (By.className (<clase elemento>))</code>	Localiza un elemento con el atributo de clase
Por nombre de etiqueta	<code>driver.findElement (By.tagName (<htmltagname>))</code>	Localiza un elemento usando la etiqueta HTML
Por texto del enlace	<code>driver.findElement (By.linkText (<LinkText>))</code>	Localiza un enlace utilizando texto del enlace
Por el texto del vínculo parcial	<code>driver.findElement (By.partialLinkText (<LinkText>))</code>	Localiza un enlace utilizando texto parcial del enlace
por CSS	<code>driver.findElement (By.cssSelector (<selector CSS>))</code>	Localiza un elemento mediante el selector CSS
por XPath	<code>driver.findElement (By.xpath (<XPath>))</code>	Localiza un elemento mediante consulta XPath

localizadores de uso

Ahora vamos a entender el uso práctico de cada uno de los métodos de localización de con la ayuda de <http://www.calculator.net>

por ID

Aquí se accede a un objeto con la ayuda de los ID. En este caso, es el ID del cuadro de texto. Los valores se introducen en el cuadro de texto con el método `sendKeys` con la ayuda de Identificación (cdensity).

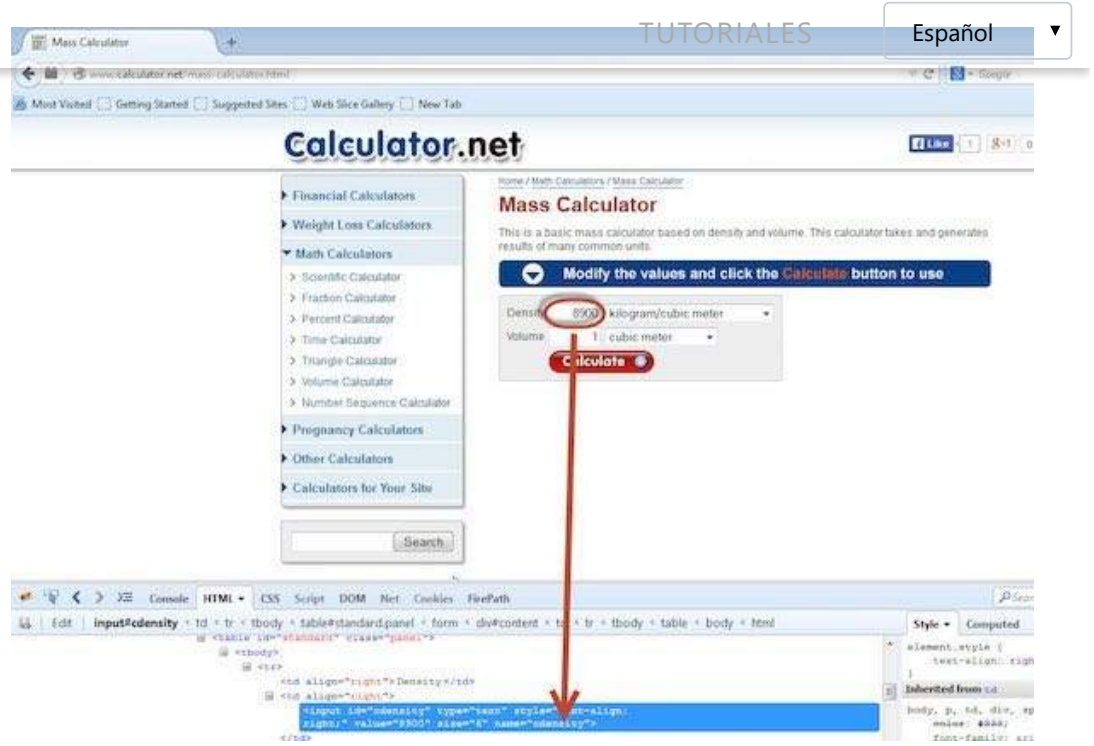
The screenshot shows the Calculator.net website with the 'Mass Calculator' selected. The interface includes input fields for 'Density' (set to 3900) and 'Volume' (set to 1), and a 'Calculate' button. A red arrow points from the '3900' value in the 'Density' field to the corresponding HTML element in the browser's developer tools. The developer tools show the HTML structure, with the following snippet highlighted:

```
<td align="right"><input id="cdensity" type="text" style="text-align: right;" value="3900" size="6" name="density"></td><td><select id="cdensityunit" name="cdensityunit"></td>
```

```
driver.findElement(By.id("cdensity")).sendKeys("10");
```

Por nombre

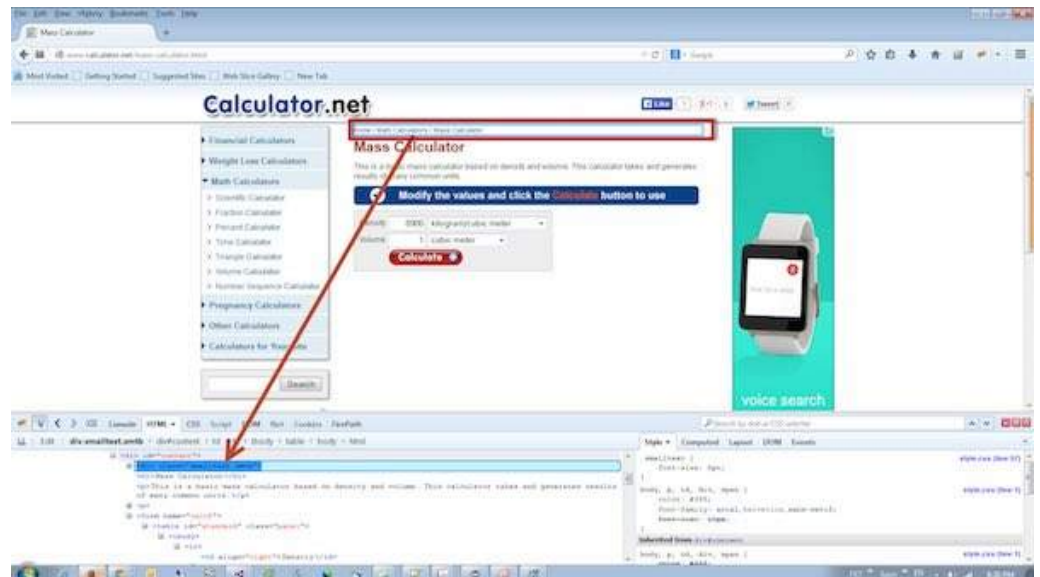
Aquí se accede a un objeto con la ayuda de nombres. En este caso, es el nombre del cuadro de texto. Los valores se introducen en el cuadro de texto con el método `sendKeys` con la ayuda de Identificación (`cdensity`).



```
driver.findElement(By.name("cdensity")).sendKeys("10");
```

Por nombre de clase

Aquí se accede a un objeto con la ayuda de nombres genéricos. En este caso, es el nombre de la clase WebElement. El valor se puede acceder con la ayuda del método getText.



```
List<WebElement> byclass = driver.findElements(By.className("smalltext smt"))
```

Por Nombre de etiqueta

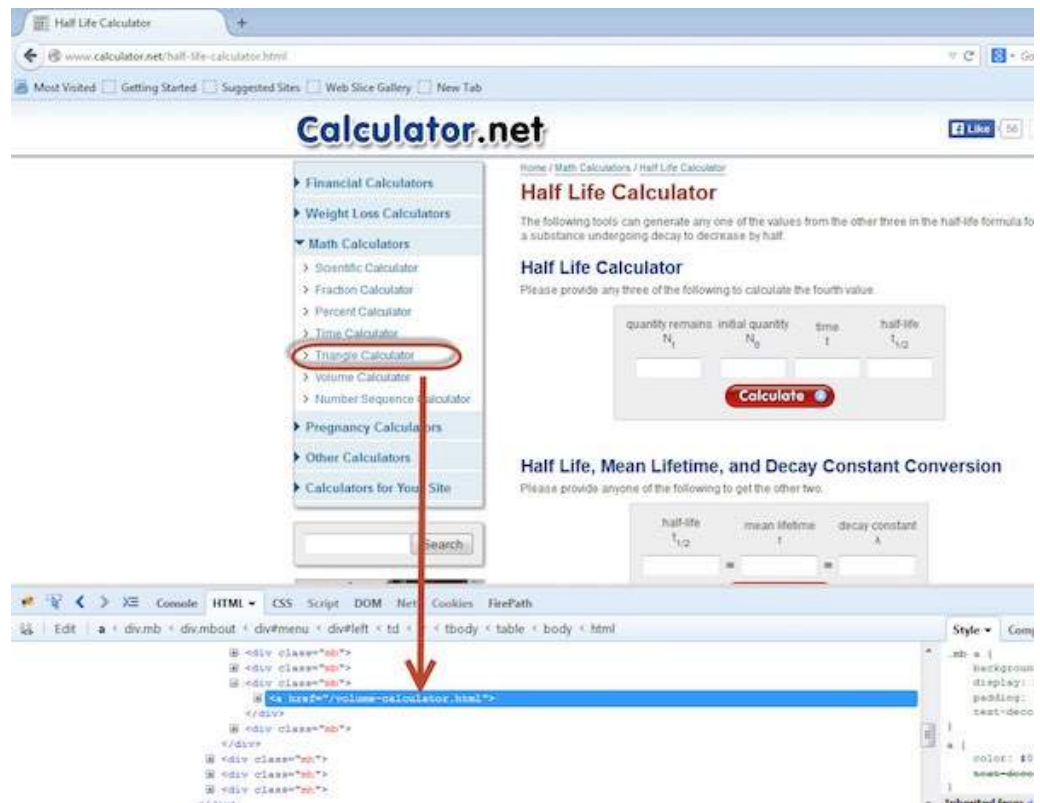


El DOM Tag nombre de un elemento puede ser usado para localizar ese elemento particular en el WebDriver. Es muy fácil de manejar tablas con la ayuda de este método. Echar un vistazo a el siguiente código.

```
WebElement table = driver.findElement(By.id("calctable"));
List<WebElement> row = table.findElements(By.tagName("tr"));
int rowcount = row.size();
```

Por Texto del enlace

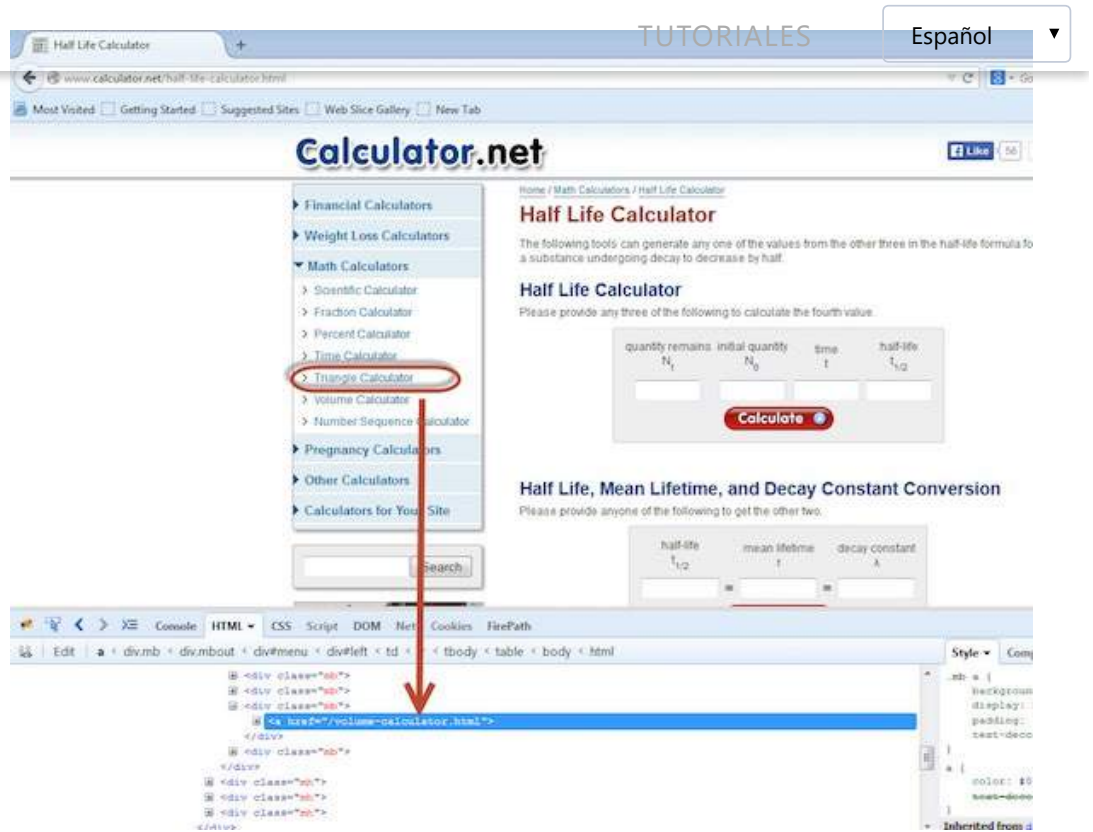
Este método ayuda a localizar un elemento de enlace con el texto a juego visibles.



```
driver.findElements(By.linkText("Volume")).click();
```

Por el texto del vínculo parcial

Este método ayuda a localizar un elemento de enlace con la concordancia parcial texto visible.



```
. Driver.findElements (By.partialLinkText ( "Volumen")) Haga clic en ();
```

por CSS

El CSS se utiliza como un método para identificar el WebObject, sin embargo, no todos los navegadores soportan la identificación CSS.

```
WebElement loginButton = driver.findElement(By.cssSelector("input.login"))
```

por XPath

XPath es sinónimo de lenguaje de ruta XML. Es un lenguaje de consulta para la selección de los nodos de un documento XML. XPath se basa en la representación del árbol de documentos XML y ofrece la posibilidad de navegar alrededor del árbol mediante la selección de nodos que utilizan una variedad de criterios.

The screenshot shows the Calculator.net website with the 'Half Life Calculator' section. The calculator has four input fields: 'quantity remains' (N_t), 'initial quantity' (N_0), 'time' (t), and 'half life' ($t_{1/2}$). A red arrow points from the 'Calculate' button to the FirePath extension in the browser's developer console. The FirePath console shows the XPath expression: `//*[@id='content']/table[1]/tbody/tr/td/table/tbody/tr[2]/td[1]/input`.

```
driver.findElement(By.xpath(".*//*[@id='content']/table[1]/tbody/tr/td/table/tbody/tr[2]/td[1]/input"))
```

Selenium - Interacciones

Las interacciones de los usuarios

WebDriver El selenio es la herramienta más utilizada entre todas las herramientas disponibles en el conjunto de herramientas de selenio. Por lo tanto, es importante entender cómo utilizar selenio para interactuar con las aplicaciones web. En este módulo, vamos a entender cómo interactuar con los objetos GUI utilizando selenio WebDriver.

Tenemos que interactuar con la aplicación mediante algunas acciones básicas o incluso alguna acción avanzada del usuario mediante el desarrollo de las funciones definidas por el usuario para el que no existen comandos predefinidos.

A continuación se enumeran los diferentes tipos de acciones en contra de esos objetos GUI:

- **Interacción Cuadro de texto (selenium_textbox.html)**
- **Botón de selección de Radio (selenium_radio_button.html)**
- **Casilla de verificación Selección (selenium_check_box.html)**
- **Menú desplegable de selección de artículos (selenium_drop_down.html)**



- [Sincronización \(selenium_synchronization.html\)](#)
- [Arrastrar y soltar \(selenium_drag_drop.html\)](#)
- [acciones del teclado \(selenium_keyboard_actions.html\)](#)
- [Las acciones del ratón \(selenium_mouse_actions.html\)](#)
- [Selección múltiple \(selenium_multi_select.html\)](#)
- [Encontrar todos los vínculos \(selenium_find_all_links.html\)](#)

Selenium - técnicas de diseño de prueba

Técnicas de diseño de la prueba de selenium

Hay varios componentes que intervienen en el diseño de las pruebas. Vamos a entender algunos de los componentes importantes que intervienen en el diseño de un marco de trabajo. Vamos a aprender los siguientes temas de este capítulo:

- [Página Object Model \(selenium_page_object_model.html\)](#)
- [Parametrización usando Excel \(selenium_parameterizing_using_excel.html\)](#)
- [log4j registro \(selenium_log4j_logging.html\)](#)
- [Manejo de excepciones \(selenium_exception_handling.html\)](#)
- [Pruebas del navegador de múltiples \(selenium_multi_browser_testing.html\)](#)
- [Imágenes de captura \(selenium_capture_screenshots.html\)](#)
- [capturar vídeos \(selenium_capture_video.html\)](#)

Selenium - TestNG

¿Cuál es TestNG?

TestNG es un potente framework de pruebas, una versión mejorada de JUnit que estaba en uso durante mucho tiempo antes de TestNG entró en existencia. GN es sinónimo de "nueva generación".

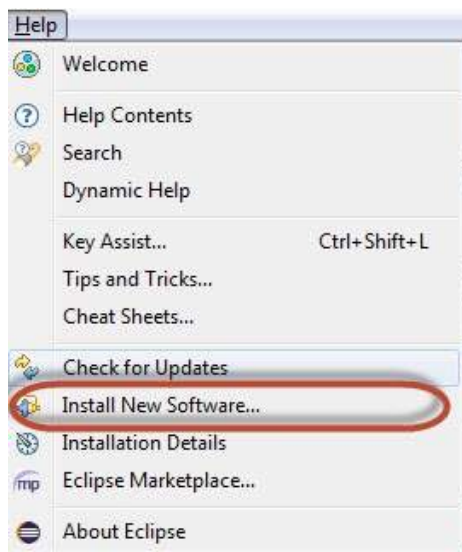
TestNG marco proporciona las siguientes características:

- Anotaciones nos ayudan a organizar las pruebas fácilmente.
- configuración de prueba flexible.
- Los casos de prueba se pueden agrupar con mayor facilidad.
- Paralelización de pruebas se puede lograr utilizando TestNG.

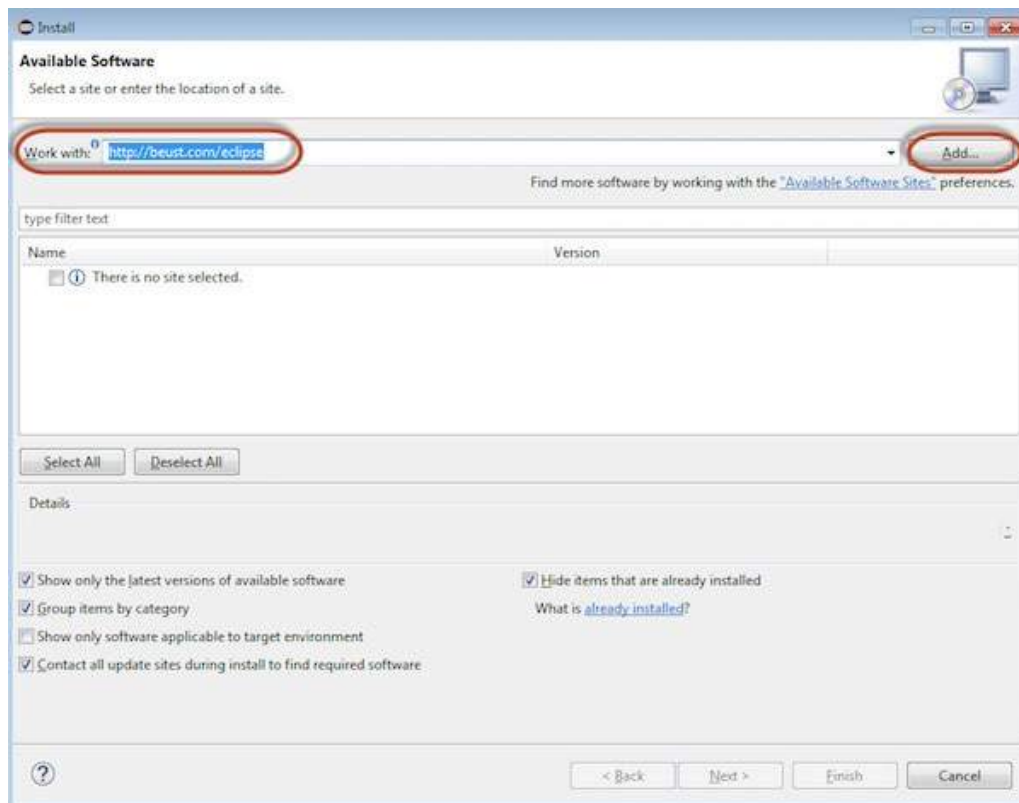
- Soporte para las pruebas basadas en datos.
- informes incorporado

Instalación de TestNG para Eclipse

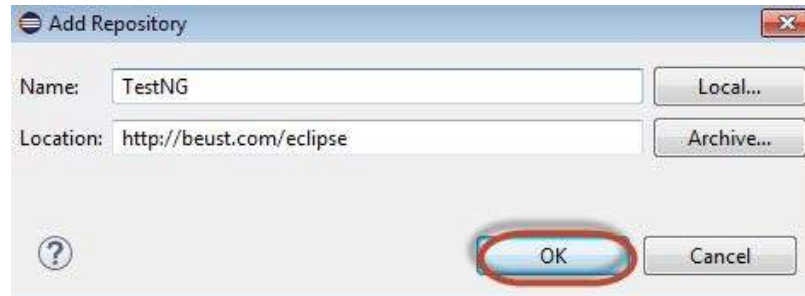
Paso 1: Lanzamiento de Eclipse y seleccione "Instalar nuevo Software".



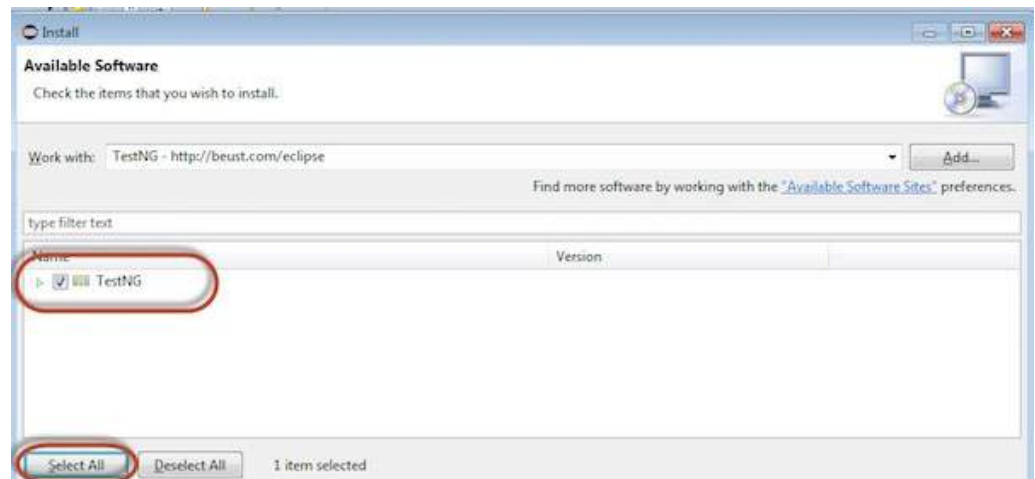
Paso 2: Introduzca la dirección URL como 'http://beust.com/eclipse' y haga clic en "Añadir".



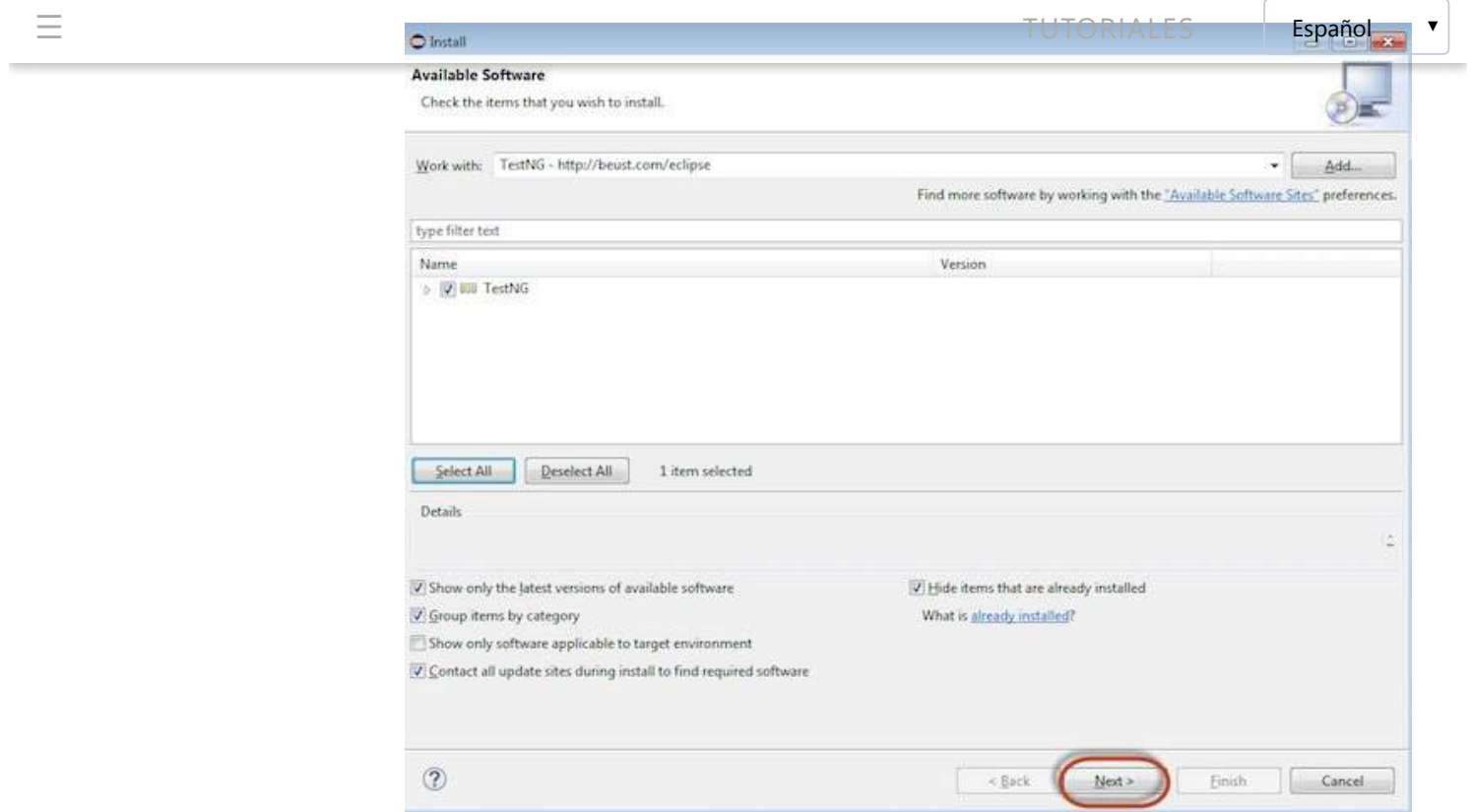
Paso 3: El cuadro de diálogo 'Añadir repositorio' se abre. Introduzca el nombre como 'TestNG' y haga clic en "OK"



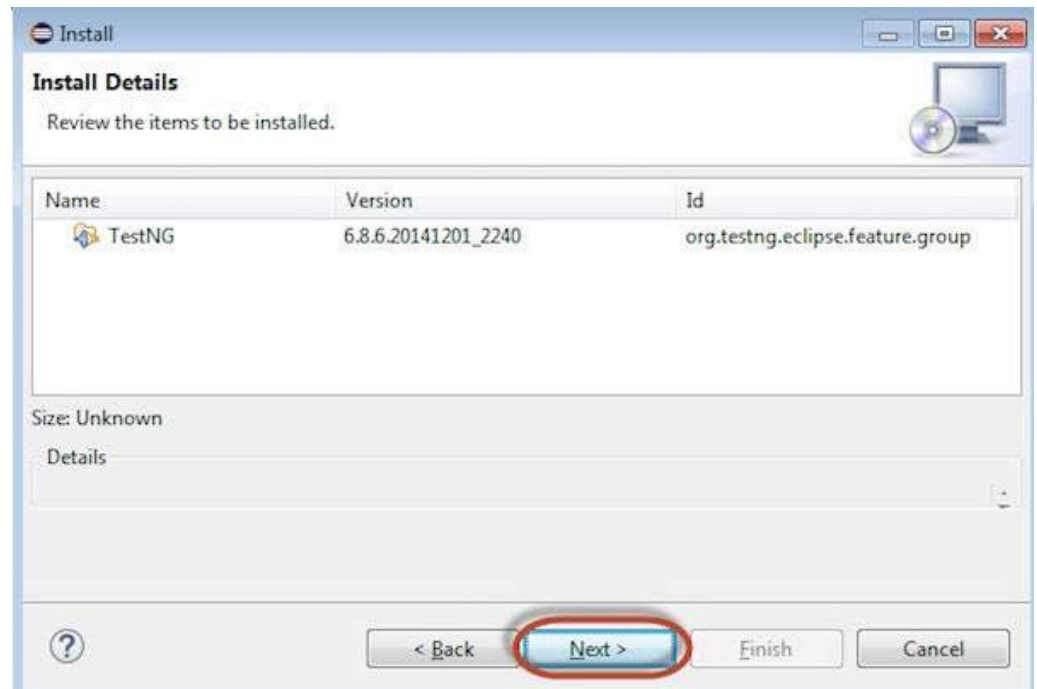
Paso 4: Haga clic en "Seleccionar todo" y "TestNG" sería seleccionado como se muestra en la figura.



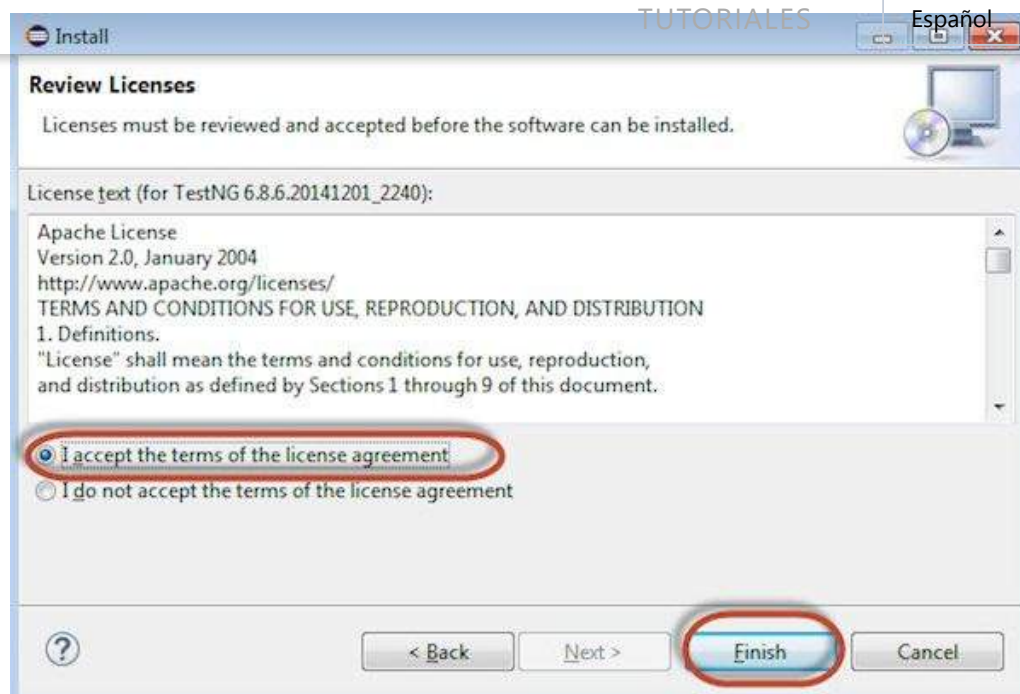
Paso 5: Haga clic en "Siguiente" para continuar.



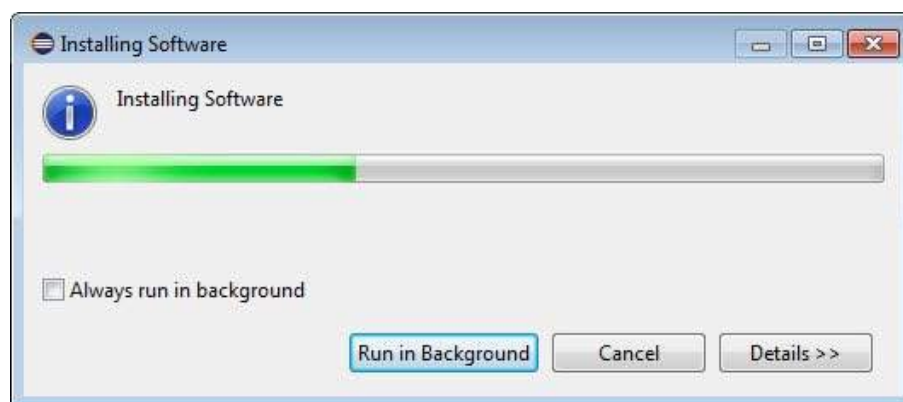
Paso 6: Revisar los elementos que están seleccionados y haga clic en 'Siguiente'.



Paso 7: "Aceptar el contrato de licencia" y haga clic en "Finalizar".



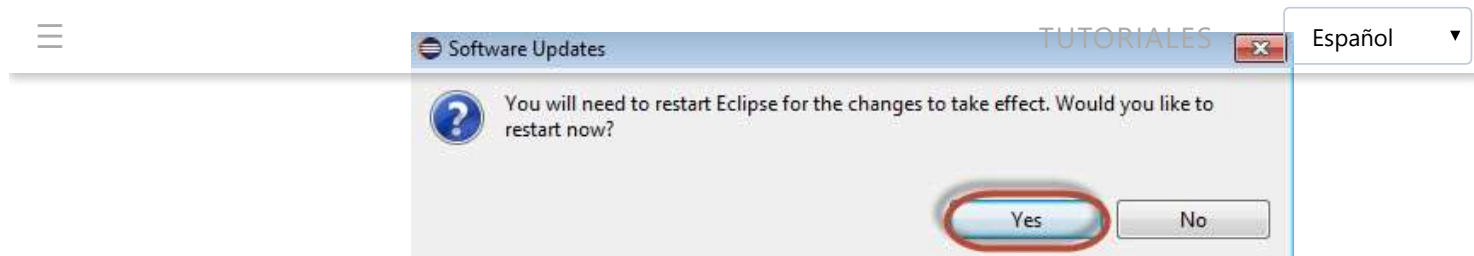
Paso 8: TestNG comienza la instalación y el progreso se muestra a continuación.



Paso 9: advertencia de seguridad aparece como no se puede establecer la validez del software. Haga clic en Aceptar'.



Paso 10: El instalador pide que reinicie Eclipse para que los cambios surtan efecto. Haga clic en "Sí".



Anotaciones en TestNG

Las anotaciones se añadieron formalmente al lenguaje Java en el JDK 5 y TestNG tomó la decisión de utilizar anotaciones para anotar las clases de prueba. Los siguientes son algunos de los beneficios del uso de anotaciones. Más información sobre TestNG se puede encontrar [aquí](http://www.w3ii.com/testng/index.htm) (<http://www.w3ii.com/testng/index.htm>)

- TestNG identifica los métodos que está interesado en por buscar anotaciones. Por lo tanto, los nombres de método no se limitan a cualquier patrón o formato.
- Podemos pasar parámetros adicionales a las anotaciones.
- Las anotaciones son fuertemente tipados, así que la bandera compilador cualquier error inmediatamente.
- Las clases de prueba ya no necesitan extenderse nada (como TestCase, para JUnit 3).

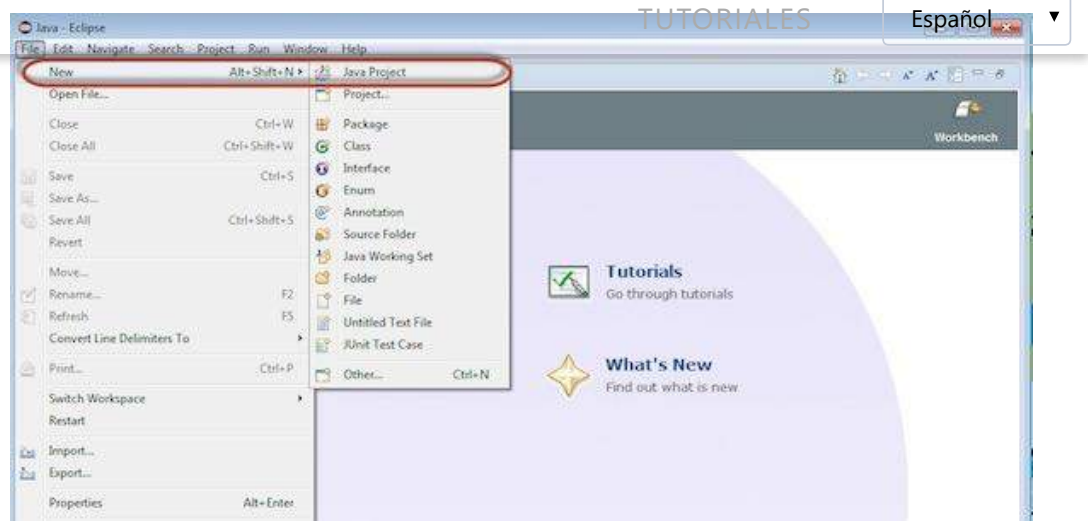
Anotación	Descripción
@BeforeSuite	El método anotado se llevará a cabo sólo una vez antes de que todas las pruebas en esta suite se hayan activado.
@AfterSuite	El método anotado se llevará a cabo sólo una vez después de todas las pruebas en esta suite se hayan activado.
@Antes de clase	El método anotado se llevará a cabo sólo una vez antes de invocar el primer método de ensayo de la clase actual.
@Después de clases	El método anotado se llevará a cabo sólo una vez después de que todos los métodos de prueba en la clase actual se han quedado.
@BeforeTest	El método anotado se llevará a cabo antes de cualquier método de ensayo pertenecen a las clases dentro de la <test> se ejecuta.
@Después de la prueba	El método anotado se llevará a cabo después de que todos los métodos de prueba que pertenecen a las clases dentro de la etiqueta <test> han corrido.



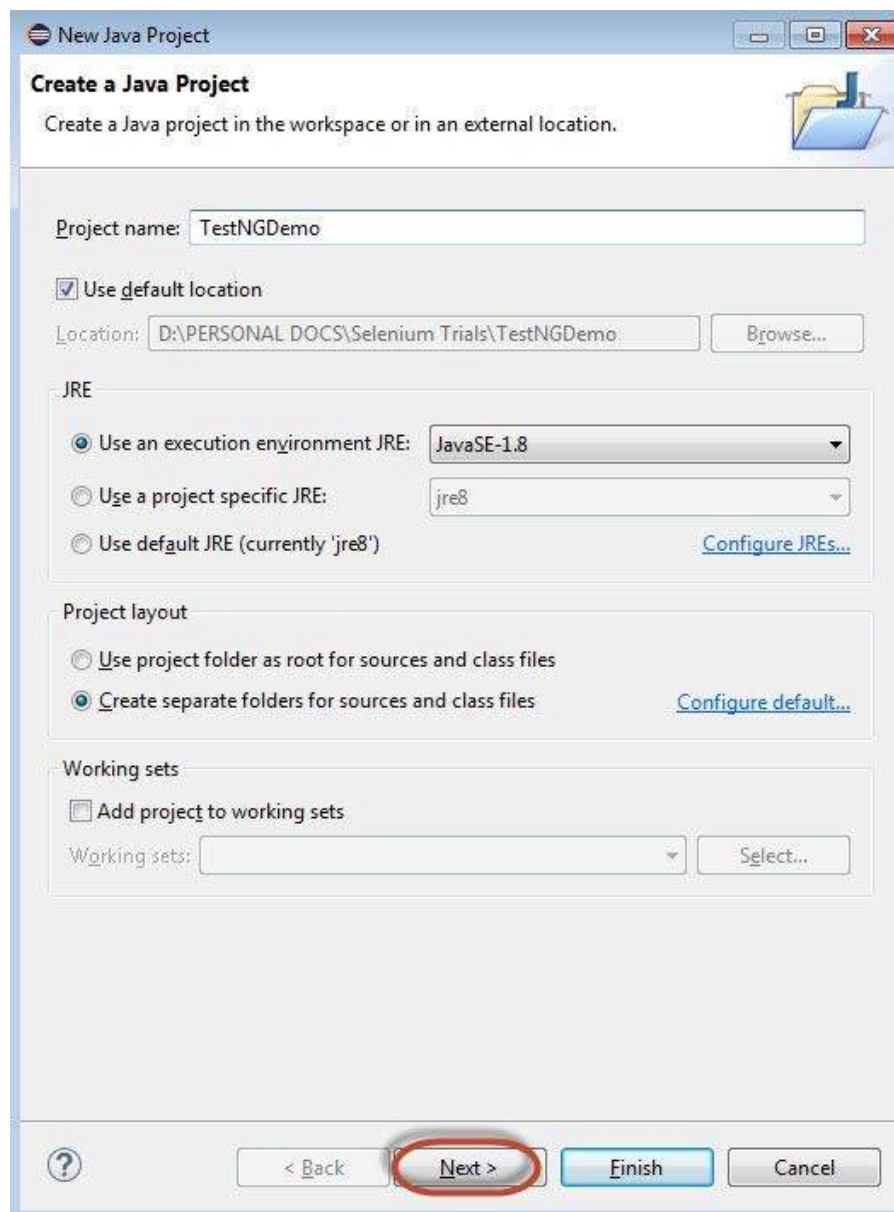
@BeforeGroups	La lista de grupos que este método de configuración se ejecutará antes. Este método se garantiza la ejecución poco antes de la primera método de ensayo que pertenece a se invoca cualquiera de estos grupos.
@AfterGroups	La lista de grupos que este método de configuración se ejecutará después. Este método se garantiza la ejecución poco después de que el último método de prueba que pertenece a se invoca cualquiera de estos grupos.
@BeforeMethod	El método anotado se llevará a cabo antes de cada método de ensayo.
@AfterMethod	El método anotado se llevará a cabo después de cada método de ensayo.
@Proveedor de datos	Marca un método como el suministro de datos para un método de ensayo. El método anotado debe devolver un Object [] [] donde cada objeto [] se puede asignar la lista de parámetros del método de ensayo. El método @test que quiere recibir los datos de este DataProvider necesita utilizar un nombre dataProvider es igual al nombre de esta anotación.
@Fábrica	Marca un método como una fábrica que devuelve objetos que serán utilizados por TestNG como clases de prueba. El método debe devolver Object [].
@Listeners	Define los oyentes en una clase de prueba.
@Parameters	Describe cómo pasar parámetros a un método @test.
@Prueba	Marca una clase o un método, como parte de la prueba.

Configuración TestNG en Eclipse

Paso 1: Iniciar Eclipse y crear un "nuevo proyecto Java ", como se muestra a continuación.

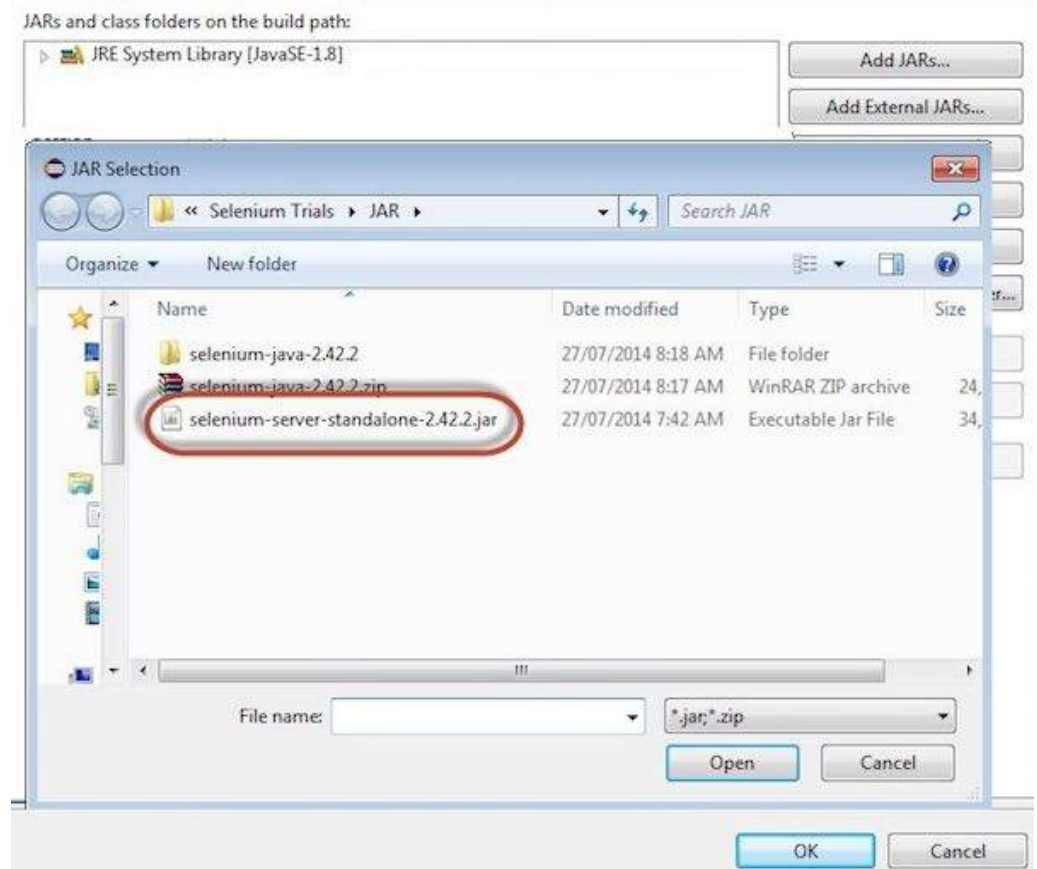


Paso 2: Introduzca el nombre del proyecto y haga clic en "Siguiete".

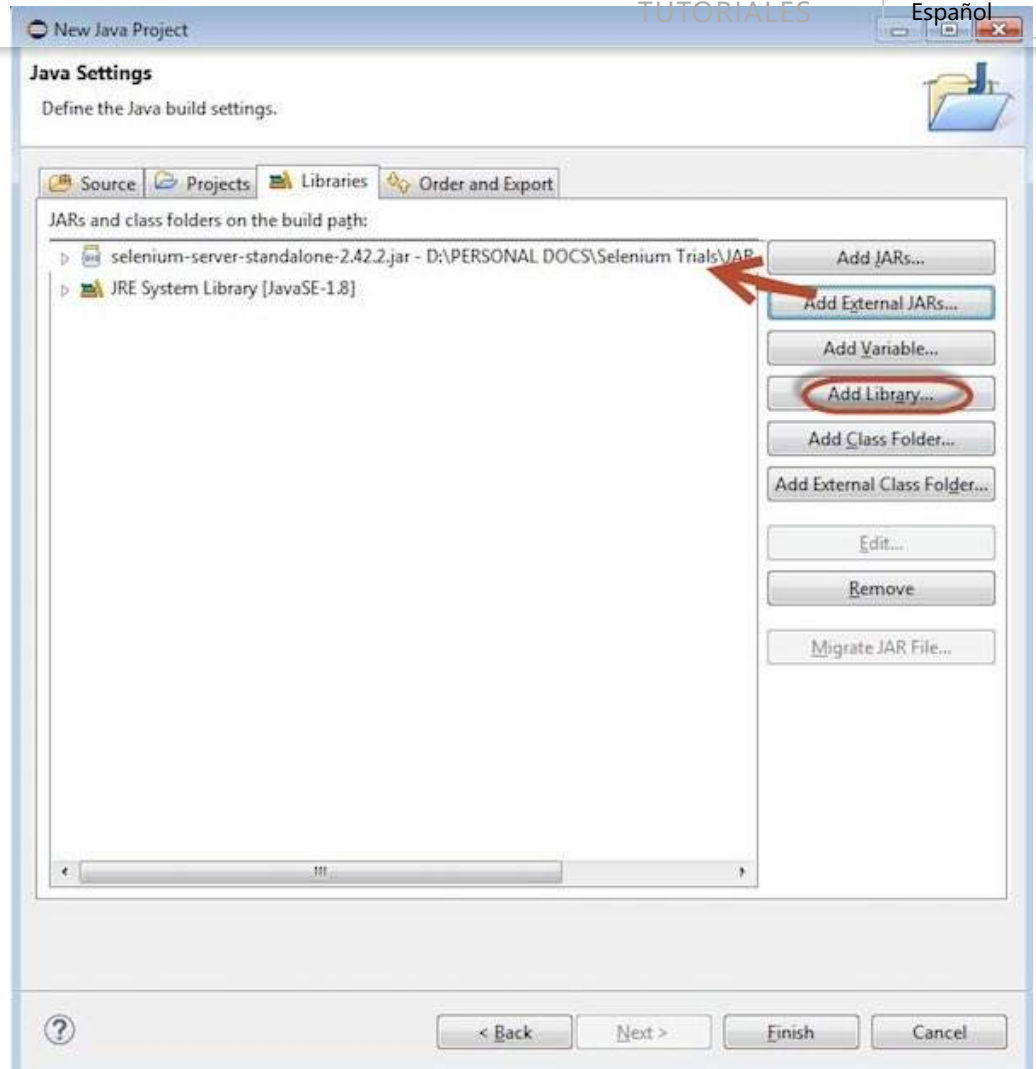




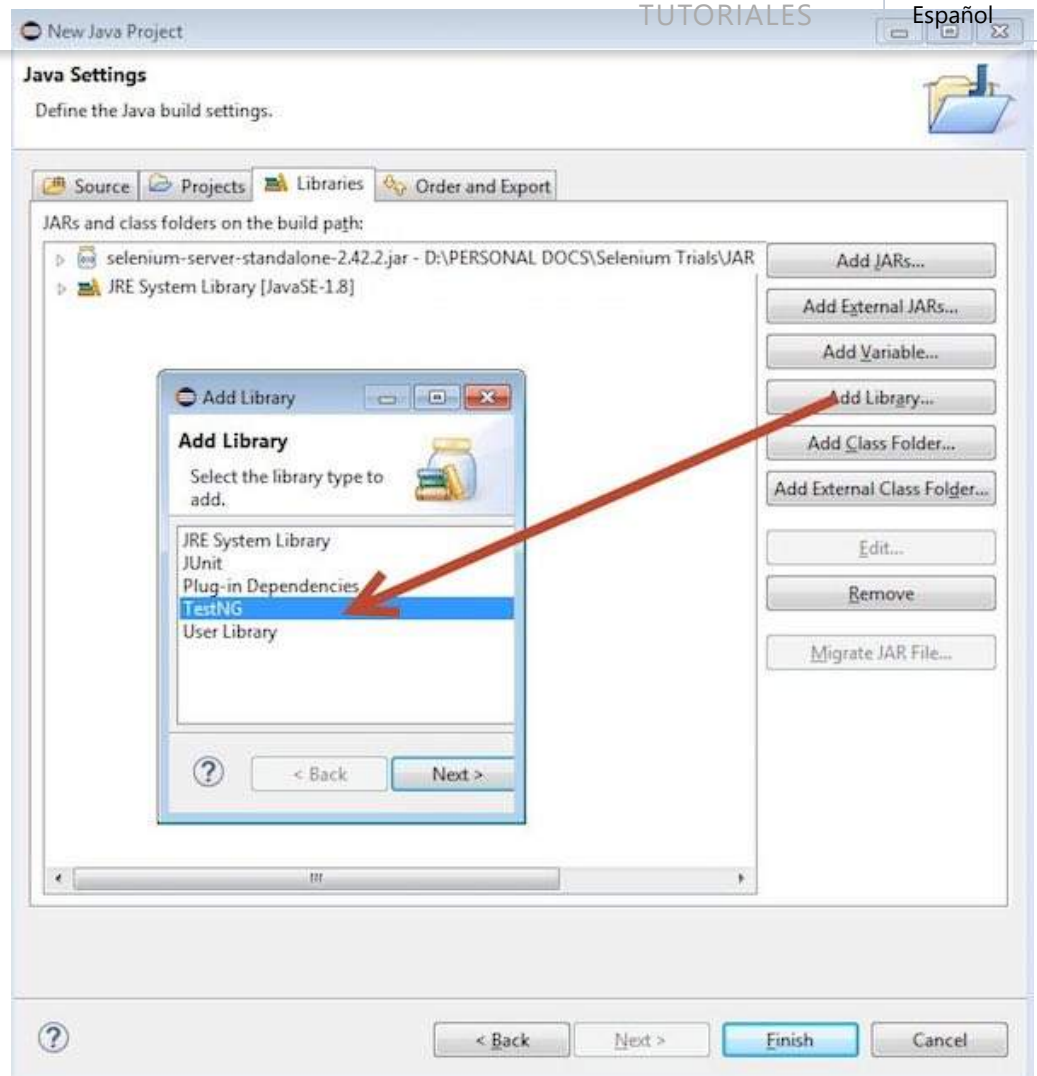
Paso 3: Vaya a "Bibliotecas" Tab y agregar el archivo JAR del servidor selenium control remoto haciendo clic en "Añadir de JAR externo" como se muestra a continuación.



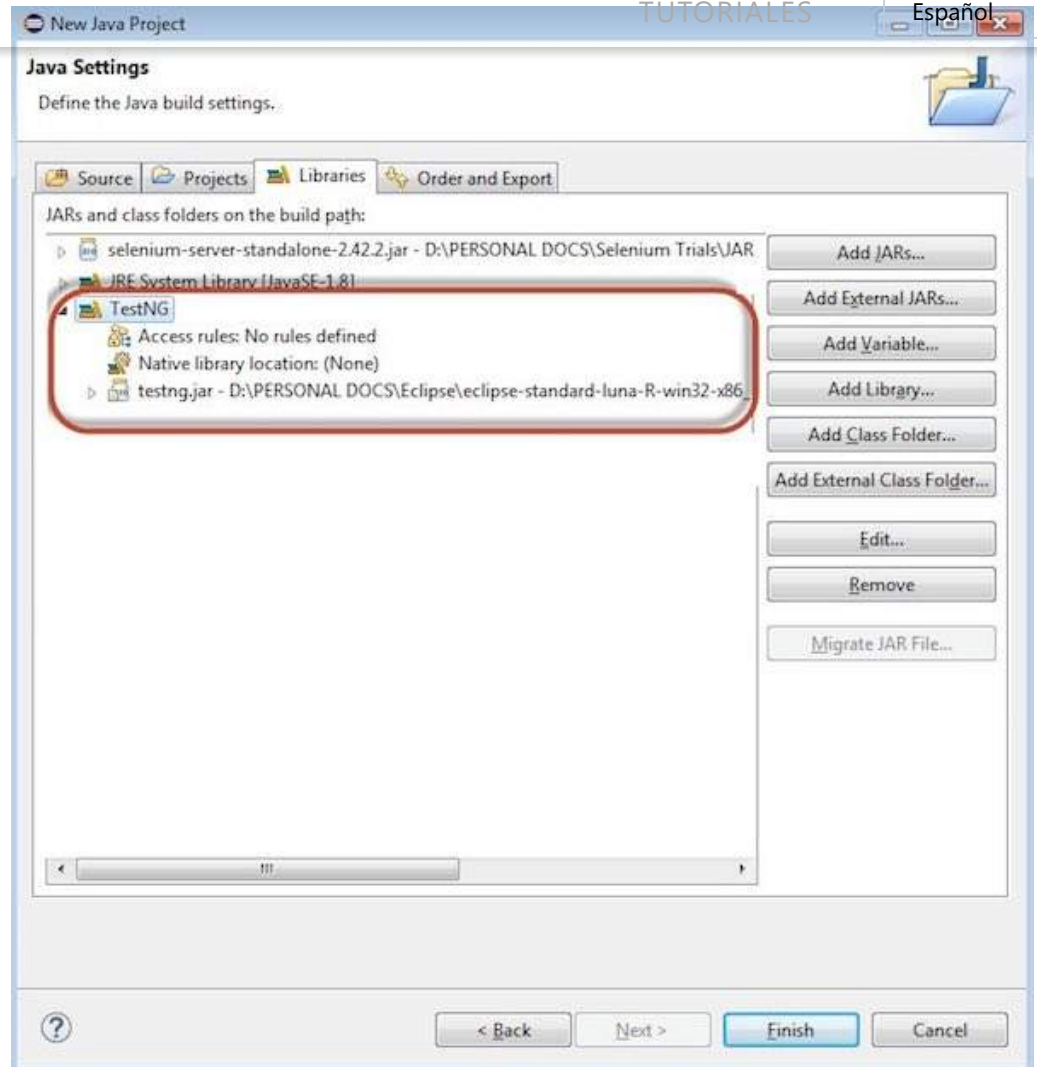
Paso 4: El archivo JAR añadido se muestra aquí. Haga clic en "Add Library '.



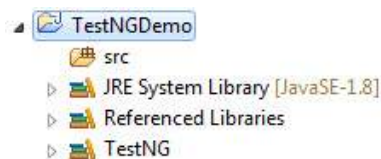
Paso 5: Se abre el diálogo 'agregar Library '. Seleccione 'TestNG' y haga clic en "Siguiete" en el cuadro de diálogo "Add Library '.



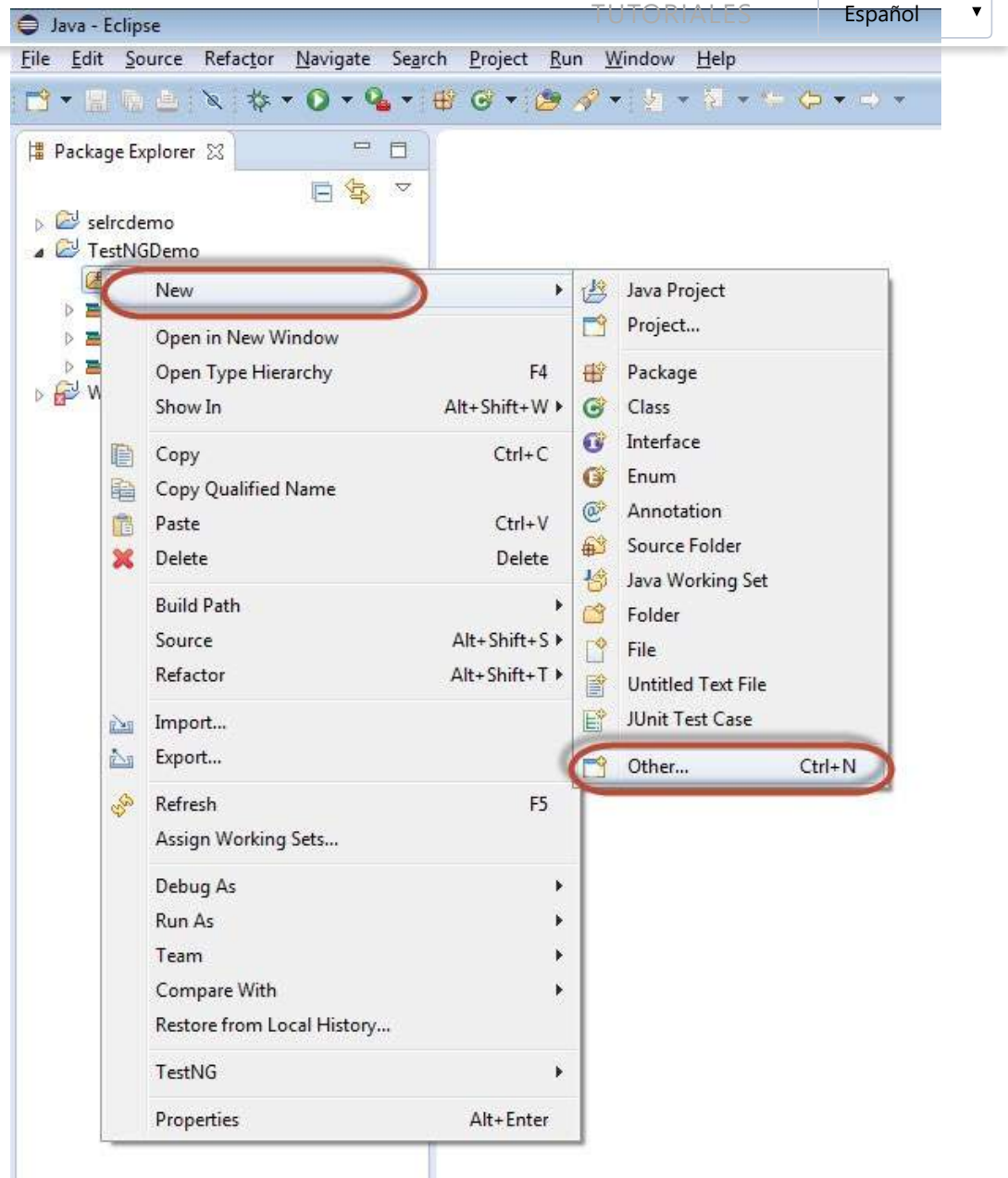
Paso 6: Se añade la Biblioteca añadido 'TestNG' y se muestra como se muestra a continuación.



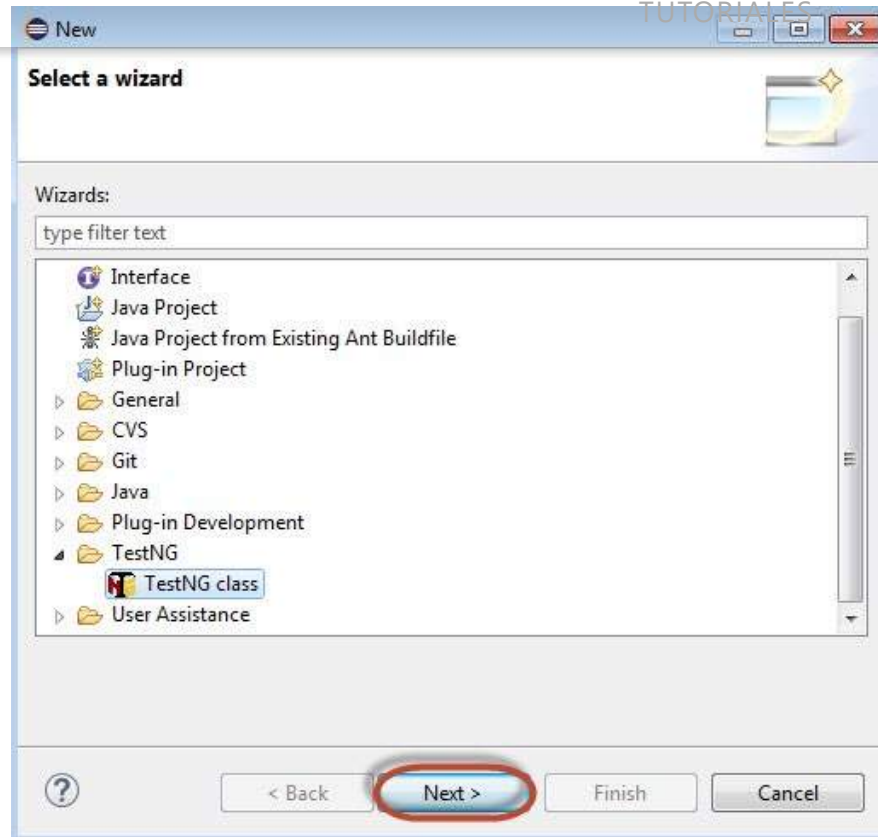
Paso 7: Después de crear el proyecto, la estructura del proyecto sería el que figura a continuación.



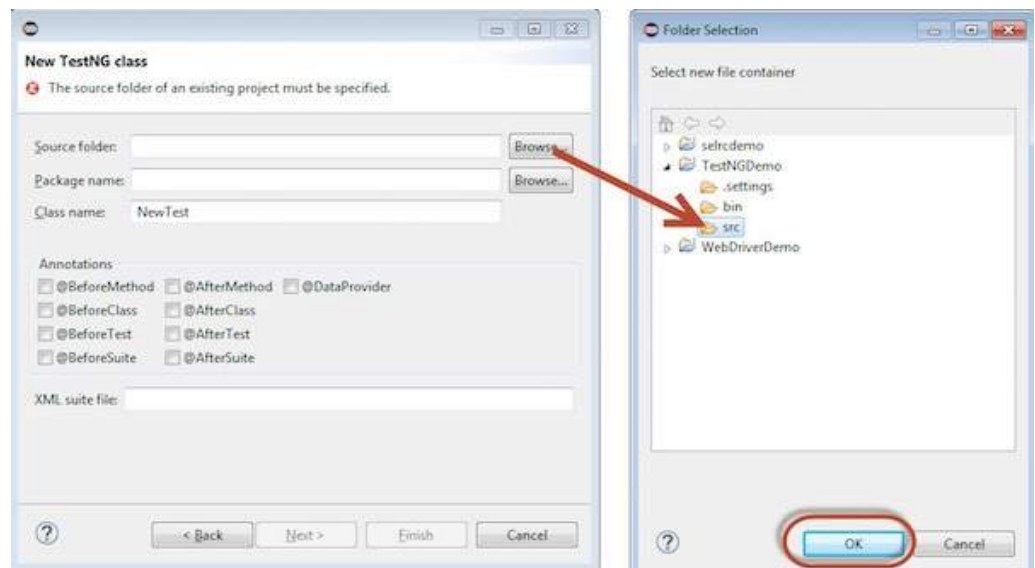
Paso 8: Haga clic en la carpeta 'src' y seleccione Nuevo >> Otro.



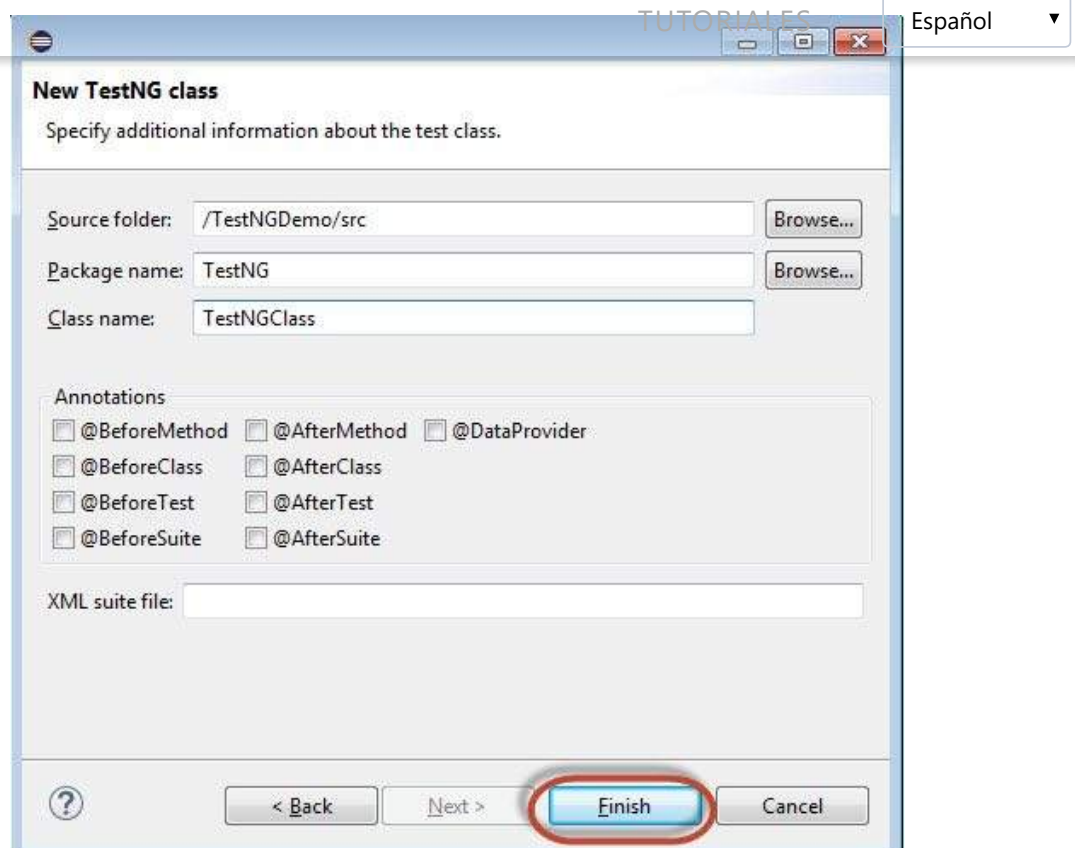
Paso 9: Seleccione 'TestNG' y haga clic en "Siguiete".



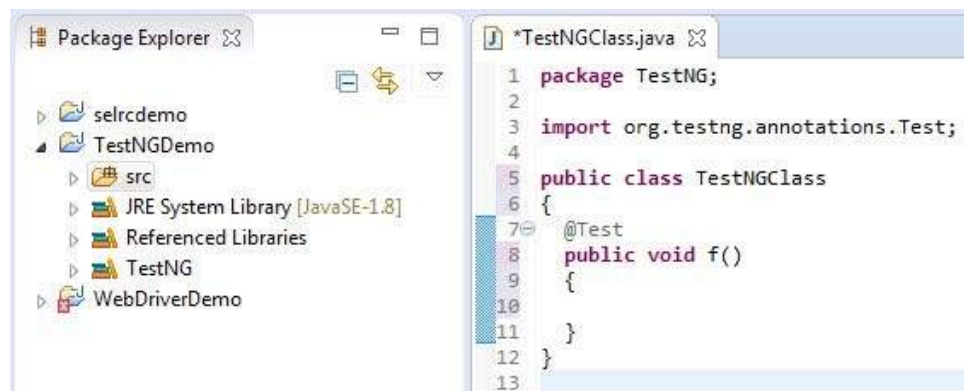
Paso 10: Seleccione el nombre de la 'Fuente de carpetas' y haga clic en "Aceptar".



Paso 11: Seleccione el nombre del paquete "", el" nombre de la clase "", y haga clic en" Finalizar "".



Paso 12: El explorador del paquete y la clase creada se muestran.



Primera prueba en TestNG

Ahora vamos a empezar de secuencias de comandos utilizando TestNG. Vamos guión para el mismo ejemplo que hemos utilizado para la comprensión de la WebDriver. Vamos a utilizar la aplicación de demostración, www.calculator.net, y llevar a cabo por ciento calculadora.

En el siguiente ensayo, se dará cuenta que no existe un método principal, como testng impulsará el flujo de ejecución del programa. Después de inicializar el conductor, se ejecutará el método '@BeforeTest' seguido de '@test' y luego



'@AfterTest'. Tenga en cuenta que no puede haber cualquier número de anotación '@test' en una clase, sino '@BeforeTest' y '@AfterTest' puede aparecer sólo una vez.

Español ▼



```
package TestNG;
import java.util.concurrent.TimeUnit;
import org.openqa.selenium.*;
import org.openqa.selenium.firefox.FirefoxDriver;
import org.testng.annotations.AfterTest;
import org.testng.annotations.BeforeTest;
import org.testng.annotations.Test;

public class TestNGClass
{
    WebDriver driver = new FirefoxDriver();

    @BeforeTest
    public void launchapp()
    {
        // Puts an Implicit wait, Will wait for 10 seconds before throwing e
        driver.manage().timeouts().implicitlyWait(10, TimeUnit.SECONDS);
        // Launch website
        driver.navigate().to("http://www.calculator.net");
        driver.manage().window().maximize();
    }
    @Test
    public void calculatepercent()
    {
        // Click on Math Calculators
        driver.findElement(By.xpath("//*[ @id='menu']/div[3]/a")).click();

        // Click on Percent Calculators
        driver.findElement(By.xpath("//*[ @id='menu']/div[4]/div[3]/a")).cli

        // Enter value 10 in the first number of the percent Calculator
        driver.findElement(By.id("cpar1")).sendKeys("10");

        // Enter value 50 in the second number of the percent Calculator
        driver.findElement(By.id("cpar2")).sendKeys("50");

        // Click Calculate Button
        driver.findElement(By.xpath("//*[ @id='content']/table/tbody/tr/td[2

        // Get the Result Text based on its xpath
        String result = driver.findElement(By.xpath("//*[ @id='content']/p[2

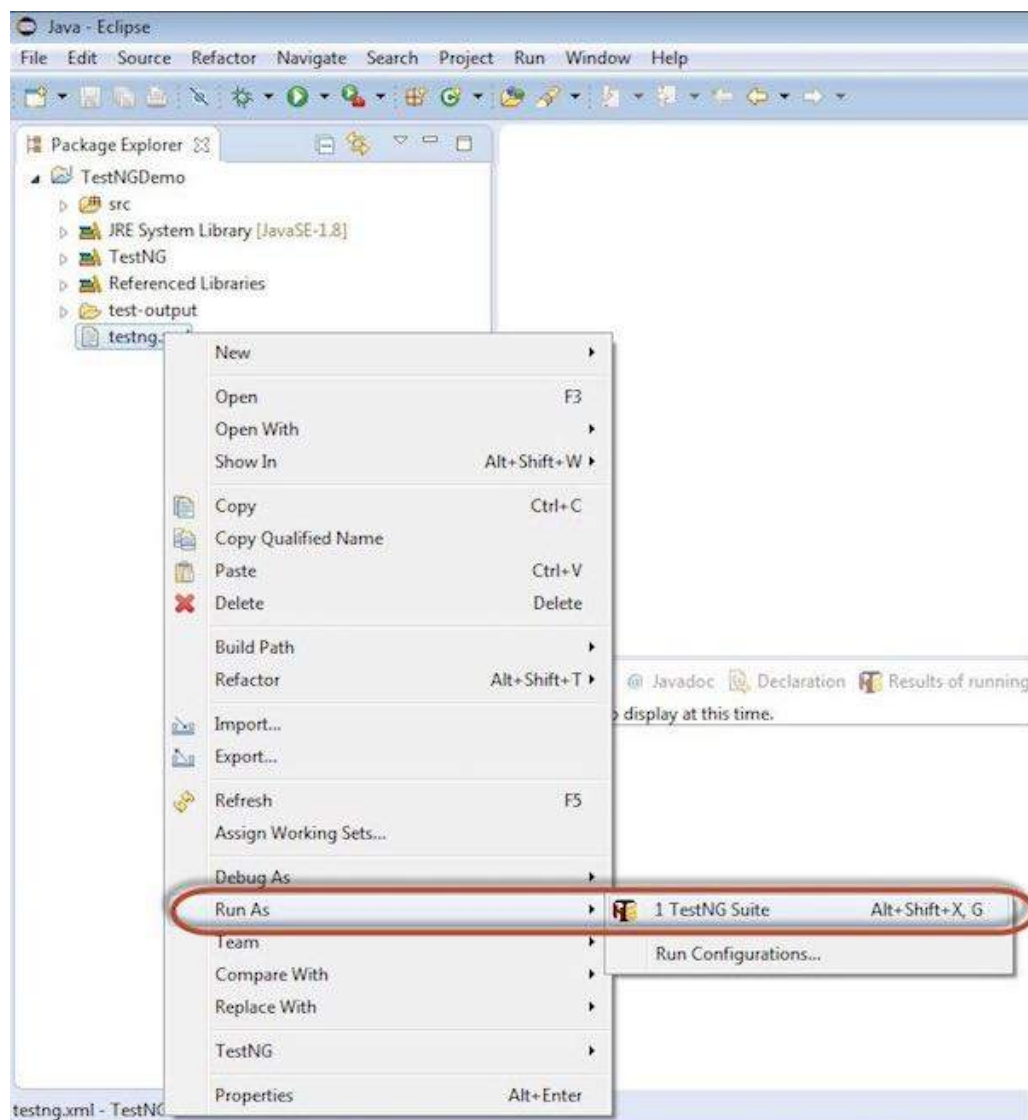
        // Print a Log In message to the screen
        System.out.println(" The Result is " + result);

        if(result.equals("5"))
        {
            System.out.println(" The Result is Pass");
        }
        else
        {
            System.out.println(" The Result is Fail");
        }
    }
}
```

```
@AfterTest
public void terminatetest()
{
    driver.close();
}
```

Ejecución

Para ejecutar, haga clic derecho en el archivo XML creado y seleccione "Ejecutar como" >> "TestNG Suite"



resultado del análisis

La salida es lanzada a la consola y que aparecería como se muestra a continuación. La salida de la consola también tiene una ejecución sumaria.



TUTORIALES

Español ▼

```
<terminated> TestNGClass [TestNG] C:\Program Files\Java\jre8\bin\javaw.exe (31 Jul 2014 8:18:51 am)
[TestNG] Running:
  C:\Users\TP\AppData\Local\Temp\testng-eclipse-1511278532\testng-customsuite.xml

  The Result is 5
  The Result is Pass
  PASSED: calculatepercent

=====

  Default test
  Tests run: 1, Failures: 0, Skips: 0
=====

Default suite
Total tests run: 1, Failures: 0, Skips: 0
=====

[TestNG] Time taken by org.testng.reporters.JUnitReportReporter@626b2d4a: 9 ms
[TestNG] Time taken by org.testng.reporters.SuiteHTMLReporter@73a8dfcc: 66 ms
[TestNG] Time taken by org.testng.reporters.EmailableReporter2@6f2b958e: 4 ms
[TestNG] Time taken by [FailedReporter passed=0 failed=0 skipped=0]: 1 ms
[TestNG] Time taken by org.testng.reporters.jq.Main@aec6354: 34 ms
[TestNG] Time taken by org.testng.reporters.XMLReporter@c2e1f26: 6 ms
```

El resultado de TestNG también se puede ver en una pestaña diferente. Haga clic en el botón "Ver informe HTML", como se muestra a continuación.



El resultado HTML se mostraría como se muestra a continuación.



Selenium - cuadrícula

El selenium cuadrícula

Rejilla de selenium es una herramienta que distribuye las pruebas a través de múltiples máquinas virtuales o físicas, para que podamos ejecutar secuencias de comandos en paralelo (simultáneamente). Se acelera drásticamente el proceso

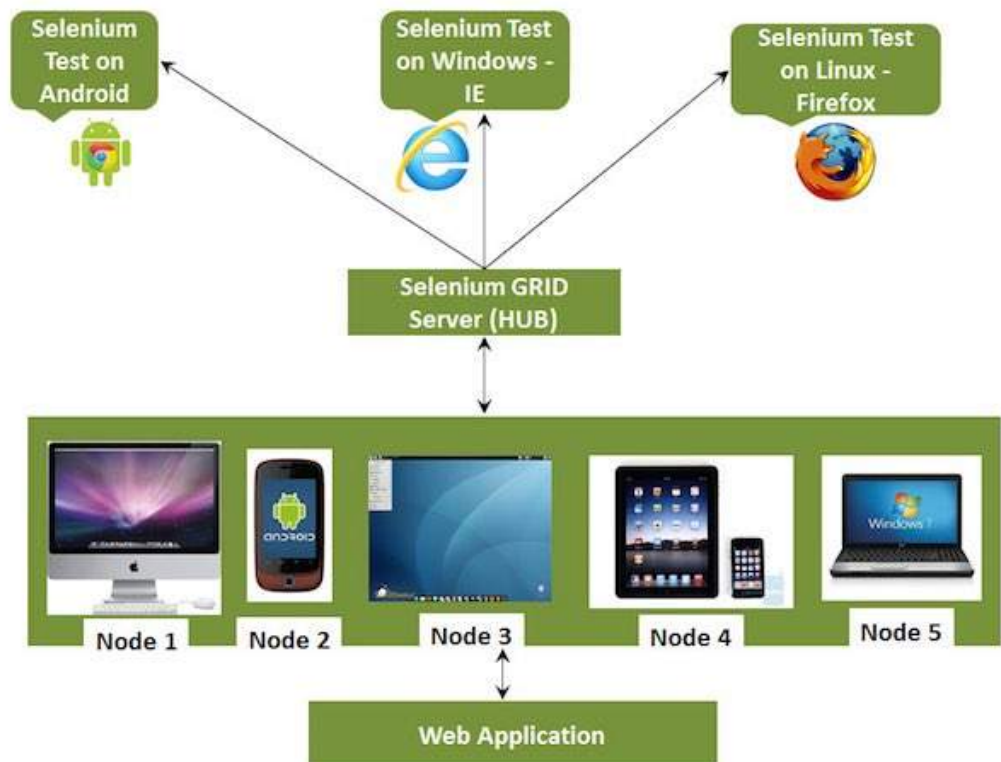
El selenium cuadrícula nos permite ejecutar varias instancias de las pruebas de control remoto WebDriver o selenium en paralelo que utiliza la misma base de código, por lo tanto, el código no necesita estar presente en el sistema que se ejecuten. El paquete de selenium de servidor autónomo incluye concentradores, WebDriver, y el selenium RC para ejecutar las secuencias de comandos en la red.

Rejilla El selenium tiene un cubo y un nodo.

- **Hub - El concentrador también puede entenderse como un servidor que actúa como el punto central donde se activarían las pruebas.** Una cuadrícula selenium tiene un solo concentrador y se puso en marcha en una sola máquina una vez.
- **- Nodos Los nodos son los casos de selenium que están conectados al concentrador que ejecutan las pruebas.** Puede haber uno o más nodos en una rejilla que puede ser de cualquier sistema operativo y pueden contener cualquiera de los navegadores de selenium compatible.

Arquitectura

El siguiente diagrama muestra la arquitectura de selenium cuadrícula.



Trabajar con cuadrícula



Con el fin de trabajar con la red, tenemos que seguir ciertos protocolos. Escuchar a continuación son los principales pasos involucrados en este proceso:

- Configuración del concentrador
- Configuración de los nodos
- Desarrollado el guión y preparar el archivo XML
- Ejecución de pruebas
- resultado del análisis

Vamos a discutir cada uno de estos pasos en detalle.

Configuración del concentrador

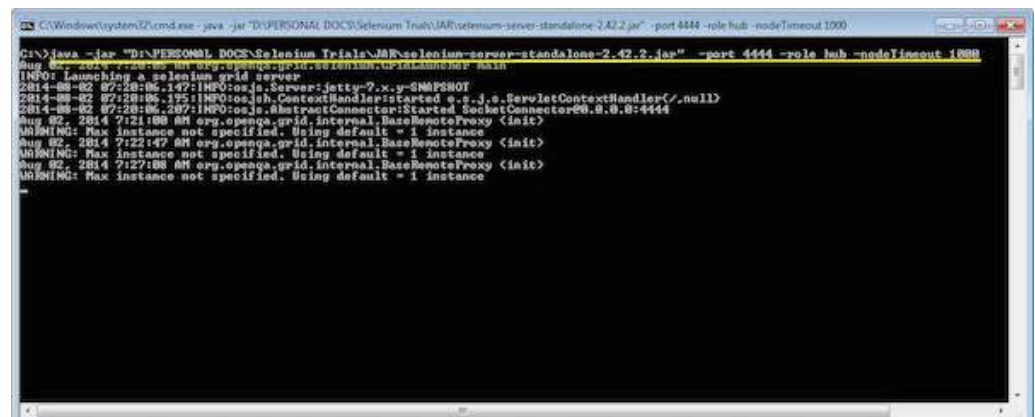
Paso 1: Descargar el último archivo JAR independiente selenio Server desde <http://docs.seleniumhq.org/download/>. Descargarlo haciendo clic en la versión que se muestra a continuación.



Paso 2: Iniciar el concentrador con el lanzamiento del selenio servidor con el siguiente comando. Ahora vamos a utilizar el puerto '4444' para iniciar el cubo.

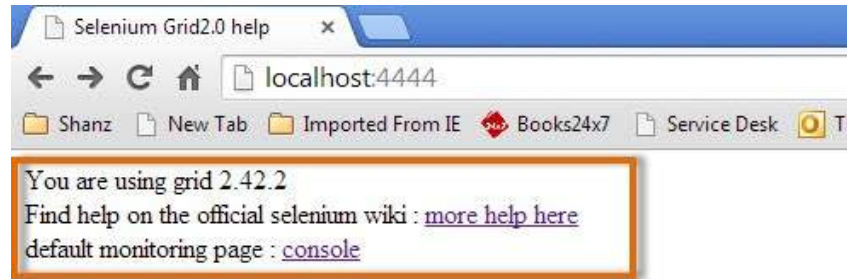
Nota: asegúrese de que no hay otras aplicaciones que se ejecutan en el puerto # 4444.

```
java -jar selenium-server-standalone-2.25.0.jar -port 4444 -role hub -node
```





Paso 3: Ahora abrir el navegador y vaya a la URL [http // localhost: 4444](http://localhost:4444) desde el Hub (El sistema en el que se ha ejecutado el paso # 2).



Paso 4: Ahora haga clic en el enlace 'consola' y haga clic en "Ver config '.

La configuración del cubo se mostrará de la siguiente manera. A partir de ahora, no hemos conseguido ningún nodo, por lo tanto, no vamos a ser capaces de ver los detalles.



Configuración de los nodos

Paso 1: iniciar sesión en el nodo (en la que desea ejecutar los scripts) y colocar el «selenium-server-standalone-2.42.2 'en una carpeta. Debemos apuntar a la JAR-selenio-servidor independiente, mientras que el lanzamiento de los nodos.

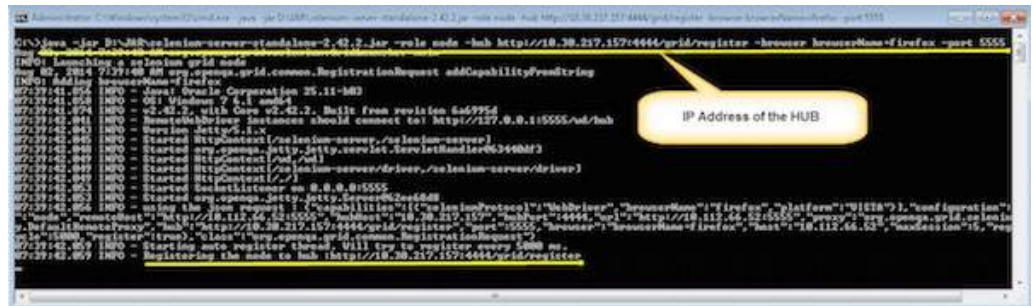
Paso 2: Lanzamiento FireFox nodo utilizando el siguiente comando a continuación.



```
java -jar D:\JAR\selenium-server-standalone-2.42.2.jar -role node -hub http://127.0.0.1:4444/wd/hub
```

Where,

```
D:\JAR\selenium-server-standalone-2.42.2.jar = Location of the Selenium Server jar
http://10.30.217.157:4444 = IP Address of the Hub and 4444 is the port of the Hub
browserName = firefox (Parameter to specify the Browser name on Nodes)
5555 = Port on which Firefox Node would be up and running.
```



Paso 3: Después de ejecutar el comando, volver al Hub. Vaya a la dirección URL - <http://10.30.217.157:4444> y el concentrador sería ahora mostrar el nodo que se le atribuye.



Paso 4: Ahora vamos a lanzar el Internet Explorer de nodo. Para el lanzamiento del Nodo IE, tenemos que tener el conductor de Internet Explorer descargado en la máquina del nodo.

Paso 5: Para descargar el controlador de Internet Explorer, vaya a <http://docs.seleniumhq.org/download/> y descargue el archivo apropiado basado en la arquitectura de su sistema operativo. Después de haber descargado, descomprimir el archivo exe y el lugar en que una carpeta que tiene que ser referido mientras que el lanzamiento de IE nodos.

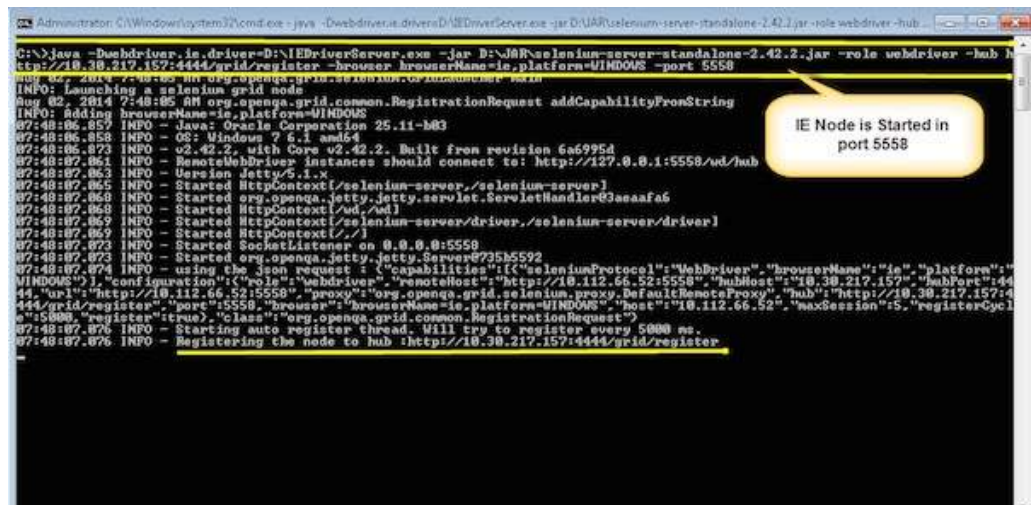


Paso 6: Poner en marcha el IE con el siguiente comando.

```
C:\>java -Dwebdriver.ie.driver=D:\IEDriverServer.exe -jar D:\JAR\selenium-
```

Where,

```
D:\IEDriverServer.exe = The location of the downloaded the IE Driver(on the
D:\JAR\selenium-server-standalone-2.42.2.jar = Location of the Selenium Se
http://10.30.217.157:4444 = IP Address of the Hub and 4444 is the port of
browserName = ie (Parameter to specify the Browser name on Nodes)
5558 = Port on which IE Node would be up and running.
```



Paso 7: Después de ejecutar el comando, volver al Hub. Vaya a la dirección URL - <http://10.30.217.157:4444> y el concentrador sería ahora mostrar el nodo IE se le atribuye.



Paso 8: Ahora vamos a lanzar Chrome Nodo. Para el lanzamiento del Nodo Chrome, tenemos que tener el conductor Chrome descargado en la máquina del nodo.

Paso 9: Para descargar el controlador de Chrome, vaya a <http://docs.seleniumhq.org/download/> y luego navegar hasta el área de navegador controladores de terceros y haga clic en el número de versión "2.10", como se muestra a continuación.

Third Party Drivers, Bindings, and Plugins

Selenium can be extended through the use of plugins. Here are a number of plugins created and maintained by third parties. For more information on how to create your own plugin or have it listed, consult the docs.

Please note that these plugins are not supported, maintained, hosted, or endorsed by the Selenium project. In addition, be advised that the plugins listed below are not necessarily licensed under the Apache License v.2.0. Some of the plugins are available under another free and open source software license; others are only available under a proprietary license. Any questions about plugins and their license of distribution need to be raised with their respective developer(s).

Third Party Browser Drivers **NOT DEVELOPED** by seleniumhq

Browser

Chrome	2.10	change log	issue tracker	selenium wiki page	Released 2014-05-01
Opera	1.5	change log	issue tracker	selenium wiki page	Released 2013-08-13
GhostDriver	(PhantomJS)		issue tracker	SeConf talk	
Windows Phone	4.14.028.10		issue tracker		Released 2013-11-23
Selendroid - Selenium for Android			issue tracker		
ios-driver			issue tracker		
BlackBerry 10			issue tracker		Released 2014-01-28
Appium			issue tracker		
Crosswalk			issue tracker		Released 2014-05-05

Paso 10: Descargue el controlador basado en el tipo de su sistema operativo. Vamos a ejecutarlo en el entorno de Windows, por lo tanto, vamos a descargar el controlador de Windows Chrome. Después de haber descargado, descomprimir el archivo exe y colocarlo en una carpeta que tiene que ser referido mientras que el lanzamiento nodos de cromo.

Index of /2.10/

Name	Last modified	Size	ETag
Parent Directory		-	
chromedriver_linux32.zip	2014-05-01 20:46:22	2.33MB	4fecc99b066cb1a346035bf022607104
chromedriver_linux64.zip	2014-05-01 22:40:58	2.20MB	058cd8b7b4b9688507701b5e648fd821
chromedriver_mac32.zip	2014-05-01 22:36:30	3.03MB	ed0dafc3ada3619edda2861f2bbaad-5c
chromedriver_win32.zip	2014-05-01 22:59:28	2.71MB	082e91e5c8994a7879710caeed62e334
notes.txt	2014-05-01 20:46:23	0.00MB	27c57f1c84c22b4427197c74f246c81a

Paso 11: Iniciar Chrome con el siguiente comando.

```
C:\>java -Dwebdriver.chrome.driver=D:\chromedriver.exe -jar D:\JAR\selenium-server-standalone-2.42.2.jar -role webdriver --hub
```

Where,

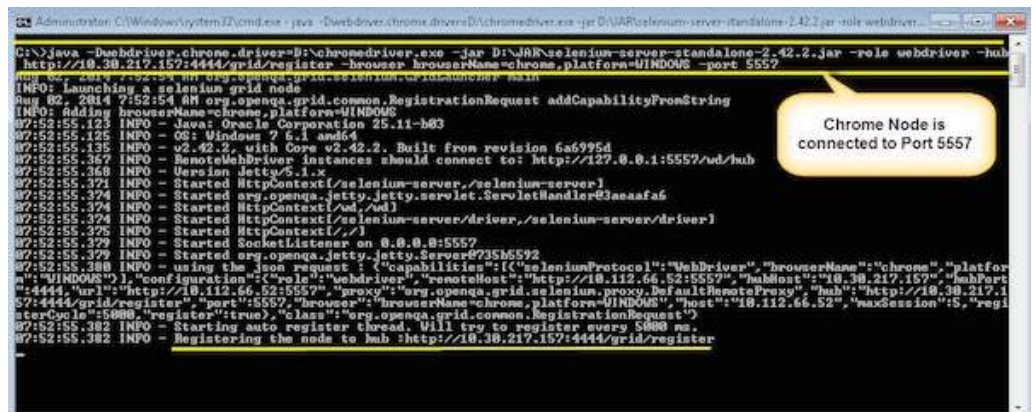
D:\chromedriver.exe = The location of the downloaded the chrome Driver(on

D:\JAR\selenium-server-standalone-2.42.2.jar = Location of the Selenium Se

http://10.30.217.157:4444 = IP Address of the Hub and 4444 is the port of

browserName = chrome (Parameter to specify the Browser name on Nodes)

5557 = Port on which chrome Node would be up and running.



Paso 12: Después de ejecutar el comando, volver al Hub. Vaya a la dirección URL - http://10.30.217.157:4444 y el concentrador sería ahora mostrar el nodo de cromo que se le atribuye.



Desarrollado el guión y preparar el archivo XML



Paso 1: Vamos a desarrollar un ensayo efectuado con TestNG. En el siguiente ejemplo, vamos a lanzar cada uno de los navegadores que utilizan WebDriver remoto. Se puede transmitir sus capacidades al conductor para que el conductor tiene toda la información para ejecutar en nodos.

El parámetro del navegador se pasa desde el archivo "XML".



```
import org.openqa.selenium.remote.DesiredCapabilities;
import java.util.concurrent.TimeUnit;
import org.openqa.selenium.*;
import org.testng.annotations.AfterTest;
import org.testng.annotations.BeforeTest;
import org.testng.annotations.Parameters;
import org.testng.annotations.Test;
import java.net.URL;
import java.net.MalformedURLException;
import org.openqa.selenium.remote.RemoteWebDriver;

public class TestNGClass
{
    public WebDriver driver;
    public String URL, Node;
    protected ThreadLocal<RemoteWebDriver> threadDriver = null;

    @Parameters("browser")
    @BeforeTest
    public void launchapp(String browser) throws MalformedURLException
    {
        String URL = "http://www.calculator.net";
        if (browser.equalsIgnoreCase("firefox"))
        {
            System.out.println(" Executing on FireFox");
            String Node = "http://10.112.66.52:5555/wd/hub";
            DesiredCapabilities cap = DesiredCapabilities.firefox();
            cap.setBrowserName("firefox");

            driver = new RemoteWebDriver(new URL(Node), cap);
            // Puts an Implicit wait, Will wait for 10 seconds before throwing
            driver.manage().timeouts().implicitlyWait(10, TimeUnit.SECONDS);

            // Launch website
            driver.navigate().to(URL);
            driver.manage().window().maximize();
        }
        else if (browser.equalsIgnoreCase("chrome"))
        {
            System.out.println(" Executing on CHROME");
            DesiredCapabilities cap = DesiredCapabilities.chrome();
            cap.setBrowserName("chrome");
            String Node = "http://10.112.66.52:5557/wd/hub";
            driver = new RemoteWebDriver(new URL(Node), cap);
            driver.manage().timeouts().implicitlyWait(10, TimeUnit.SECONDS);

            // Launch website
            driver.navigate().to(URL);
            driver.manage().window().maximize();
        }
        else if (browser.equalsIgnoreCase("ie"))
        {

```



```

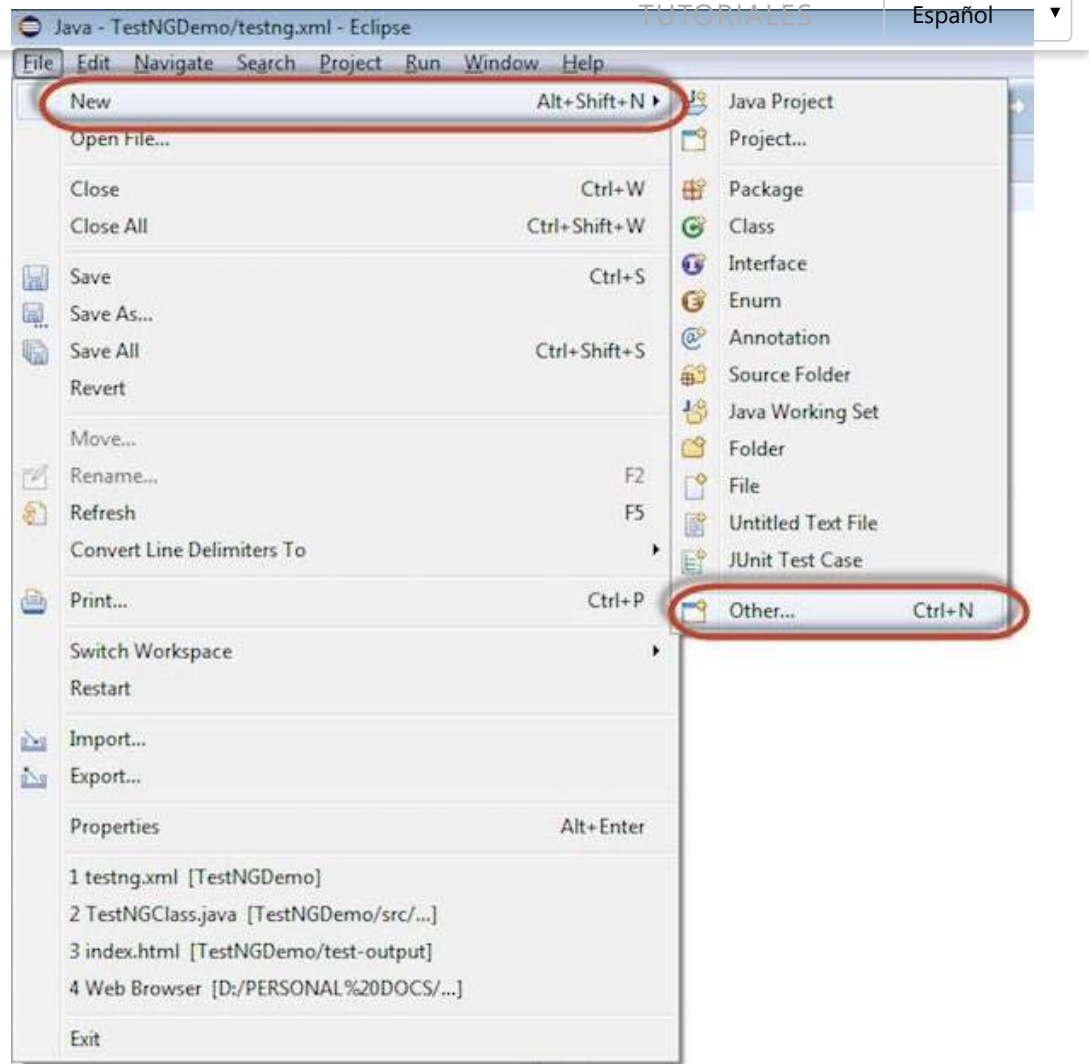
System.out.println(" Executing on IE");
DesiredCapabilities cap = DesiredCapabilities.chrome();
cap.setBrowserName("ie");
String Node = "http://10.112.66.52:5558/wd/hub";
driver = new RemoteWebDriver(new URL(Node), cap);
driver.manage().timeouts().implicitlyWait(10, TimeUnit.SECONDS);

// Launch website
driver.navigate().to(URL);
driver.manage().window().maximize();
}
else
{
    throw new IllegalArgumentException("The Browser Type is Undefined");
}
}
@Test
public void calculatepercent()
{
    // Click on Math Calculators
    driver.findElement(By.xpath(".*[@id='menu']/div[3]/a")).click();
    // Click on Percent Calculators
    driver.findElement(By.xpath(".*[@id='menu']/div[4]/div[3]/a")).click();
    // Enter value 10 in the first number of the percent Calculator
    driver.findElement(By.id("cpar1")).sendKeys("10");
    // Enter value 50 in the second number of the percent Calculator
    driver.findElement(By.id("cpar2")).sendKeys("50");

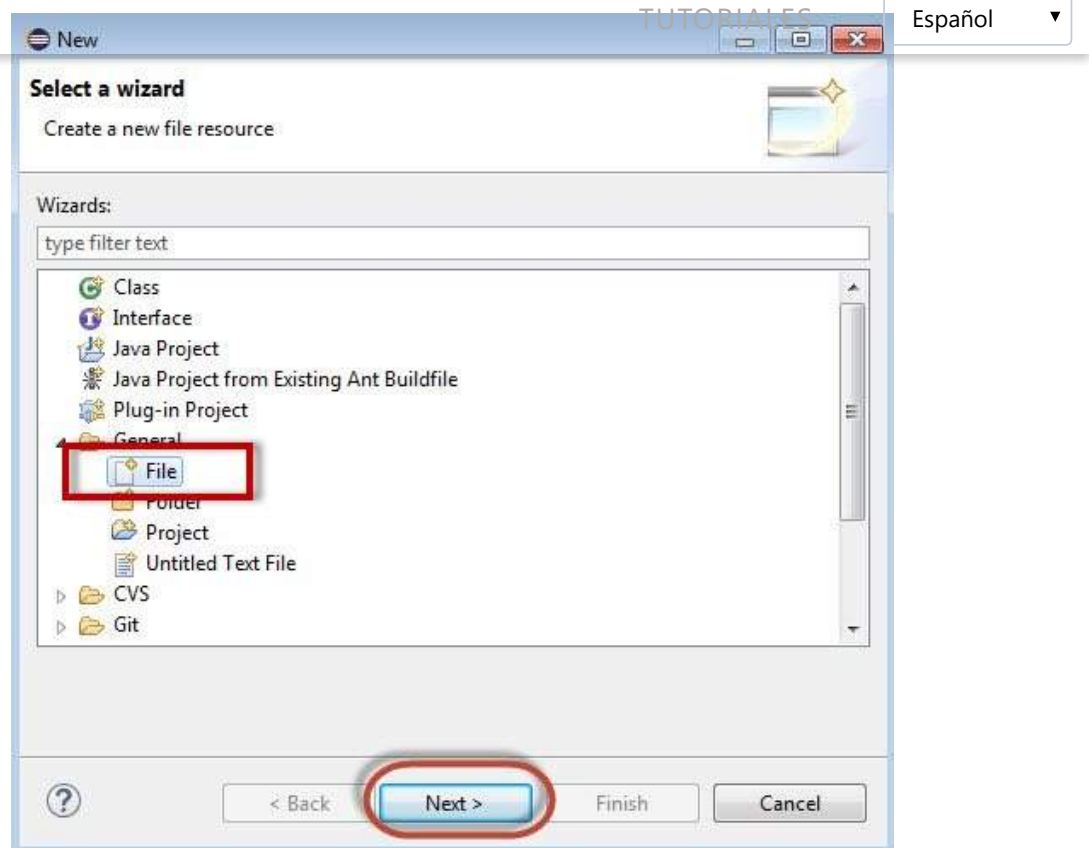
    // Click Calculate Button driver.findElement(By.xpath(".*[@id='calculate']"));
    // Get the Result Text based on its xpath
    String result = driver.findElement(By.xpath(".*[@id='content']/p[2]")).getText();
    // Print a Log In message to the screen
    System.out.println(" The Result is " + result);
    if(result.equals("5"))
    {
        System.out.println(" The Result is Pass");
    }
    else
    {
        System.out.println(" The Result is Fail");
    }
}
@AfterTest
public void closeBrowser()
{
    driver.quit();
}
}

```

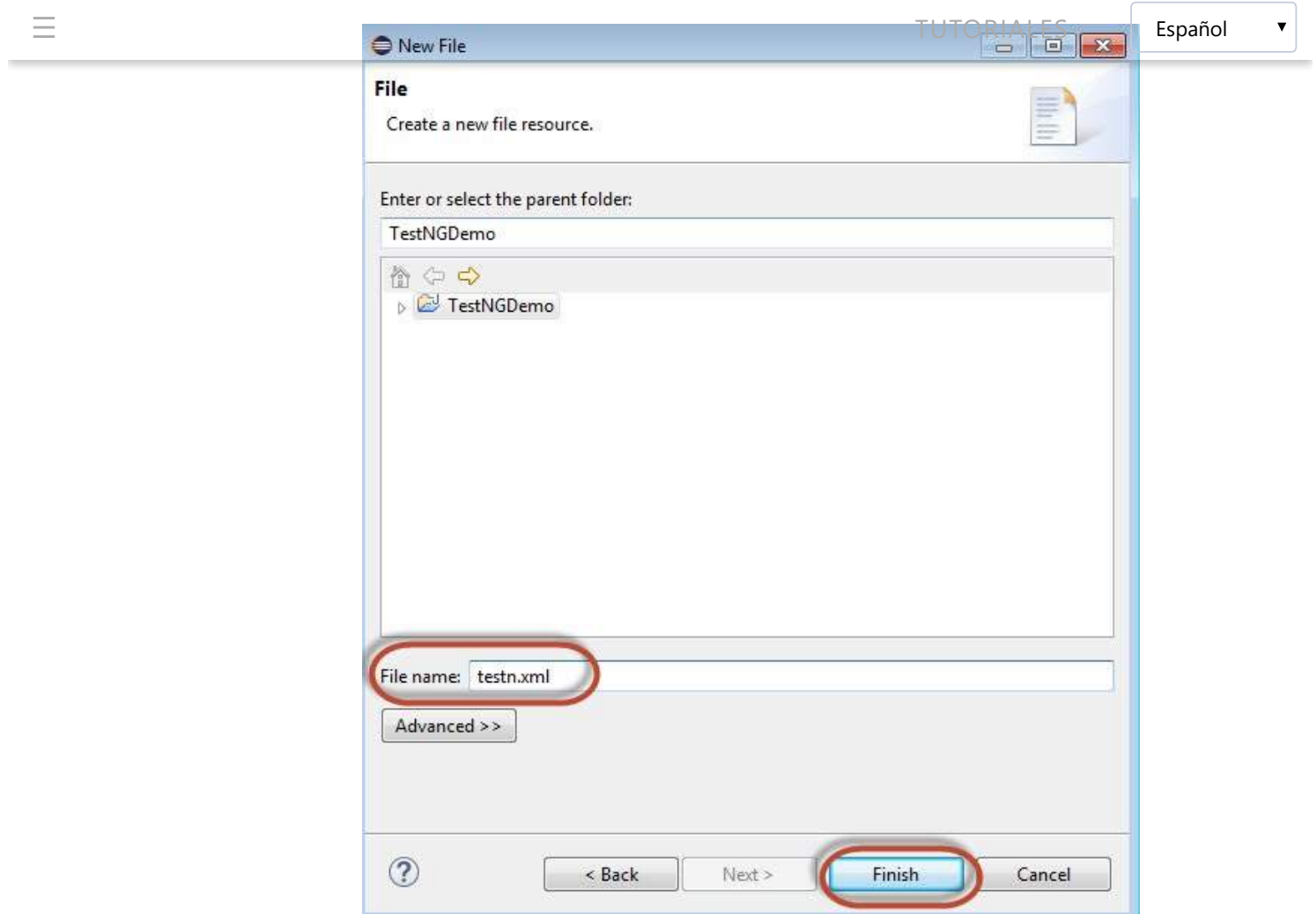
Paso 2: El parámetro Browser se pasará el uso de XML. Crear un archivo XML en la carpeta de proyecto.



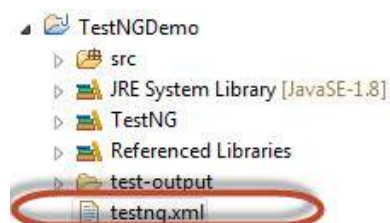
Paso 3: Seleccione "Archivo" de "general" y haga clic en "Siguiente".



Paso 4: Introduzca el nombre del archivo y haga clic en "Finalizar".



Paso 5: testng.xml se crea en la carpeta del proyecto como se muestra a continuación.



Paso 6: El contenido del archivo XML se muestran a continuación.

Creamos 3 pruebas y los puso en una suite y mencionamos = "pruebas" paralelas, de modo que todas las pruebas se ejecutan en paralelo.



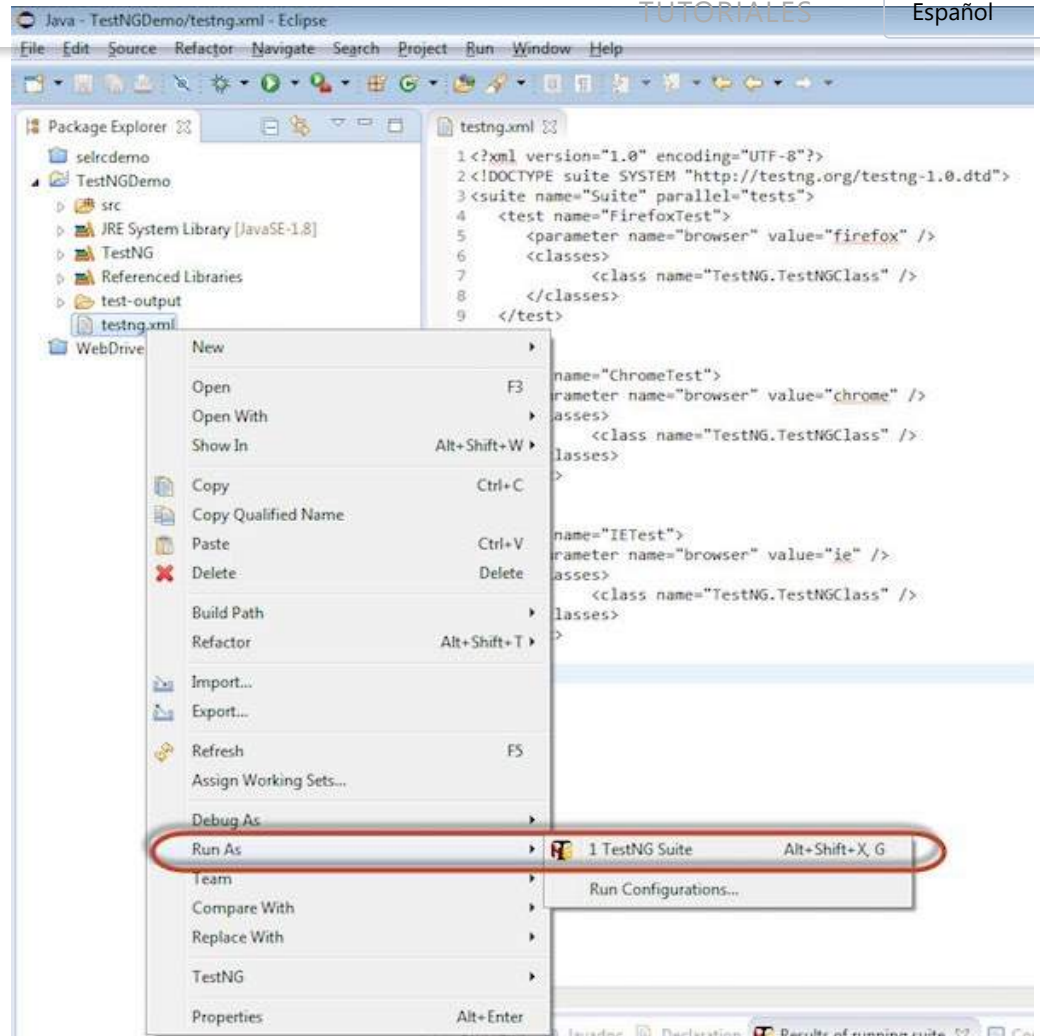
```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE suite SYSTEM "http://testng.org/testng-1.0.dtd">
<suite name="Suite" parallel="tests">
  <test name="FirefoxTest">
    <parameter name="browser" value="firefox" />
    <classes>
      <class name="TestNG.TestNGClass" />
    </classes>
  </test>

  <test name="ChromeTest">
    <parameter name="browser" value="chrome" />
    <classes>
      <class name="TestNG.TestNGClass" />
    </classes>
  </test>

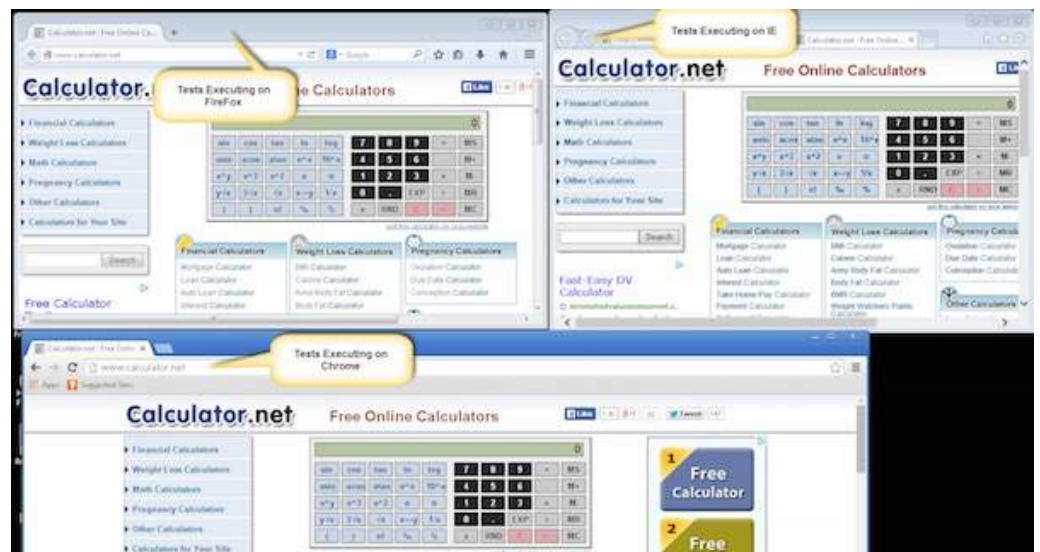
  <test name="IETest">
    <parameter name="browser" value="ie" />
    <classes>
      <class name="TestNG.TestNGClass" />
    </classes>
  </test>
</suite>
```

Ejecución de pruebas

Paso 1: Seleccionar el XML creado; derecho del ratón y seleccione "Ejecutar como" >> "TestNG Suite".



Paso 2: Ahora abra el nodo, en el que hemos puesto en marcha todos los nodos del navegador. Verá todos los tres navegadores en ejecución al mismo tiempo.



resultado del análisis



Paso 1: Al término de la ejecución, podemos analizar el resultado como cualquier otra ejecución. El resumen de los resultados se imprime en la consola como se muestra en la siguiente instantánea.

```

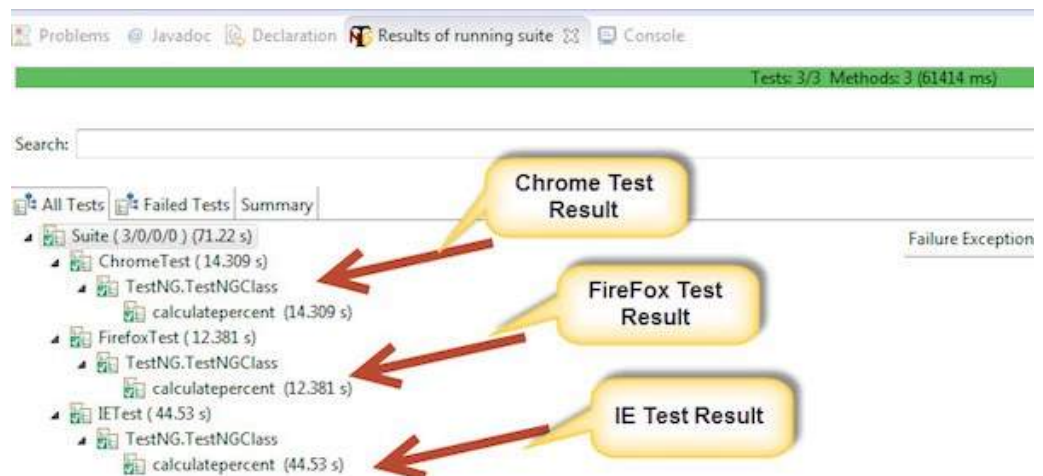
Problems @ Javadoc Declaration Results of running suite Console
<terminated> testng.xml [TestNG] C:\Program Files\Java\jre8\bin\javaw.exe (2 Aug 2014 8:25:04 am)
[TestNG] Running:
  D:\PERSONAL DOCS\Selenium Trials\TestNGDemo\testng.xml

Executing on FireFox
Executing on IE
Executing on CHROME
The Result is 5
The Result is Pass
The Result is 5
The Result is Pass
The Result is 5
The Result is Pass

=====
Suite
Total tests run: 3, Failures: 0, Skips: 0
=====

```

Paso 2: Vaya a los resultados de la ejecución 'Suite' Tab y TestNG se mostrará el resumen de los resultados como se muestra a continuación.



Paso 3: Al generar el código HTML, vamos a ser capaces de ver los resultados de las pruebas en formato HTML.



TUTORIALES

Español ▼

Test results

1 suite

All suites

Suite

Info

- D:\PERSONAL DOCS\Selenium Trials\TestNGDemo\testing.xml
- 3 tests
- 0 groups
- Times
- Reporter output
- Ignored methods
- Chronological view

Results

- 3 methods, 3 passed
- Passed methods (hide)
- ☒ calculatepercent
- ☒ calculatepercent
- ☒ calculatepercent

Tests for Suite

FirefoxTest (1 class)

ChromeTest (1 class)

IETest (1 class)

< Anterior (selenium_grids.html)

Siguiente capítulo > (selenium_useful_resources.html)

Enlaces patrocinados

[Inicio \(http://www.w3ii.com/es/index.html\)](http://www.w3ii.com/es/index.html) [HTML Tutorial \(http://www.w3biq.com/es/\)](http://www.w3biq.com/es/) [Bootstrap \(http://www.bootstrap1.com/es/\)](http://www.bootstrap1.com/es/) [Foro \(http://www.w3iz.com\)](http://www.w3iz.com) [That is America \(http://en-US.that-is-america.com\)](http://en-US.that-is-america.com)

©2017 W3im.com. All Rights Reserved. Email: jacklee8902@gmail.com

