

# **Selenium**

**Laboratorio 05**  
**Henry Mendoza Puerta**

# INTRODUCCIÓN

- Conjunto de herramientas para automatizar pruebas funcionales sobre sistemas basados en web a través de diferentes plataformas.





### Selenium Web Driver

Puede ejecutar un script en un navegador de forma nativa, sin necesidad de usar Selenium Server.



### Selenium IDE

Es un plugin para Firefox, que permite grabar un script y reproducirlo.



### Selenium Remote Control

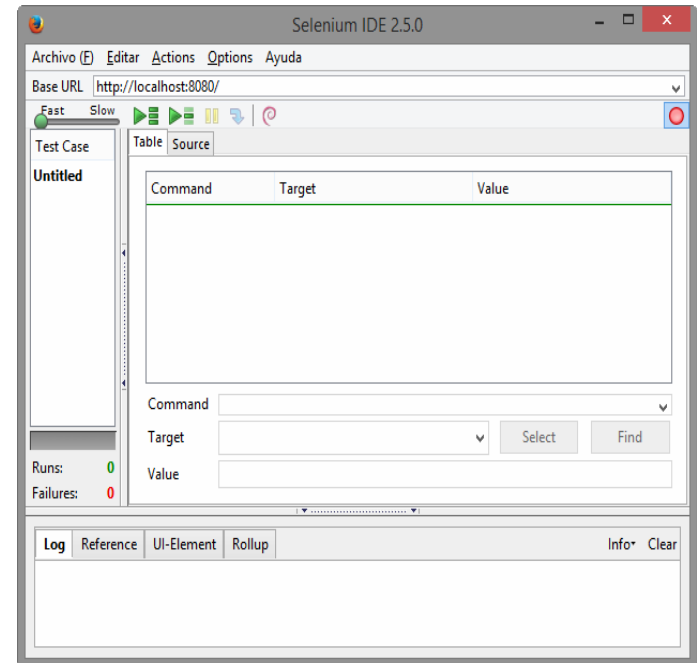
Es una aplicación de tipo cliente/servidor que permite ejecutar aplicaciones remotas, usando el Selenium Server.



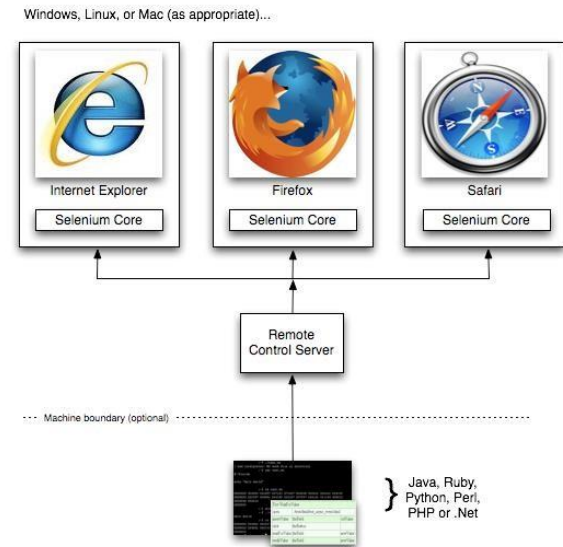
### Selenium Grid

Puede ejecutar aplicaciones remotas de manera concurrente.

- Plug in de Firefox cuya finalidad es grabar flujos de prueba de manera fácil y rápida.
- Se puede descargar e instalar en Firefox.



- Se trata de un API (Application Programming Interface) que permite programar pruebas automatizadas de sistemas web en determinados lenguajes de programación (Ruby, Java, C#, etc)
- Su arquitectura se basa en un sistema cliente-servidor, con este se puede ejecutar pruebas sobre IE, FF y Safari.



- Se trata de la versión mejorada de Selenium RC.
- Crear tests que se ejecuten sin necesidad de un server a través de drivers.

- Se trata de una implementación de Web Driver, de modo que los tests se puedan ejecutar en paralelo, utilizando poder computacional de una o mas computadoras.

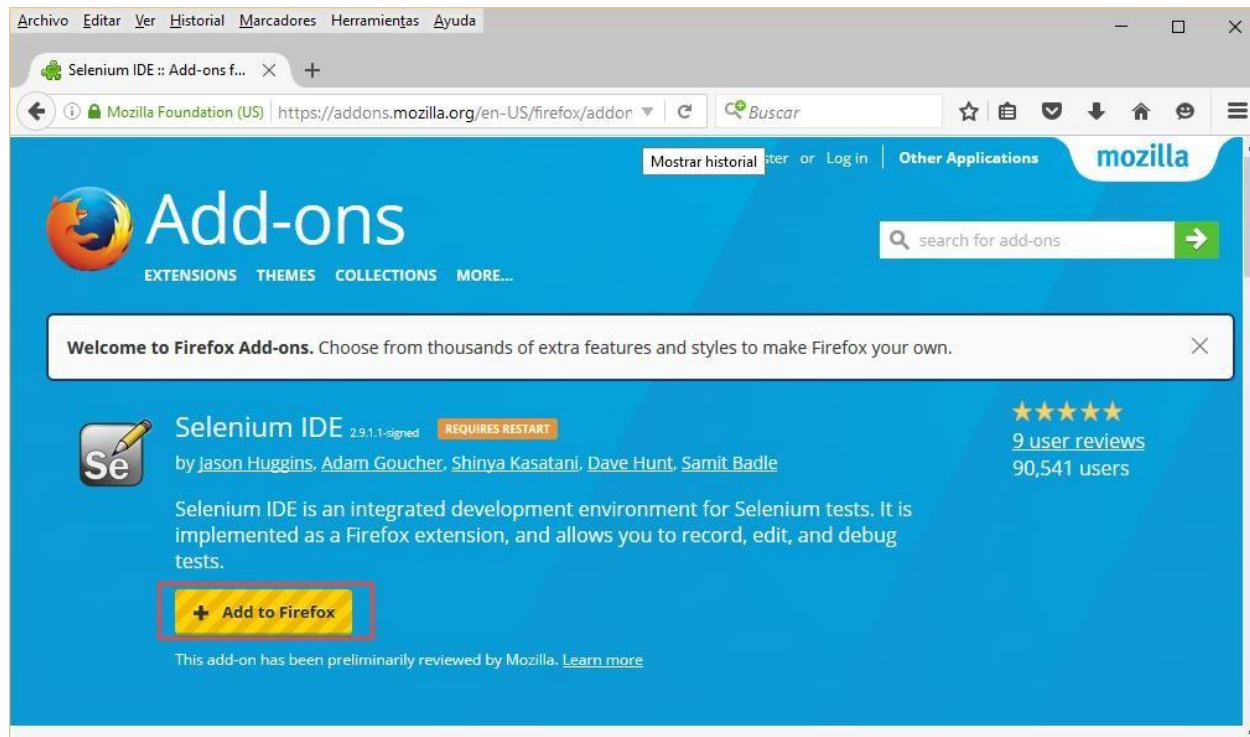


SELENIUM IDE

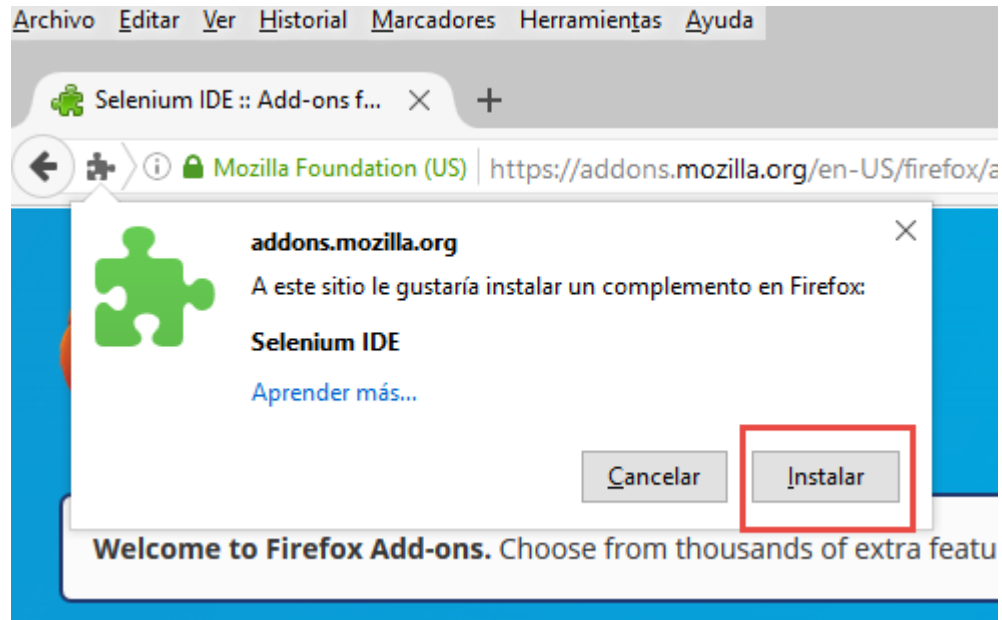
- Entorno de desarrollo integrado utilizado para
  - programar casos de prueba.
- Curva corta de aprendizaje
- Cuenta con opciones de menú contextual para grabar comandos (o líneas de código) a través de la Interface de Usuario.

Descargar el instalador del Selenium IDE de la ruta:

<https://addons.mozilla.org/en-US/firefox/addon/selenium-ide/>

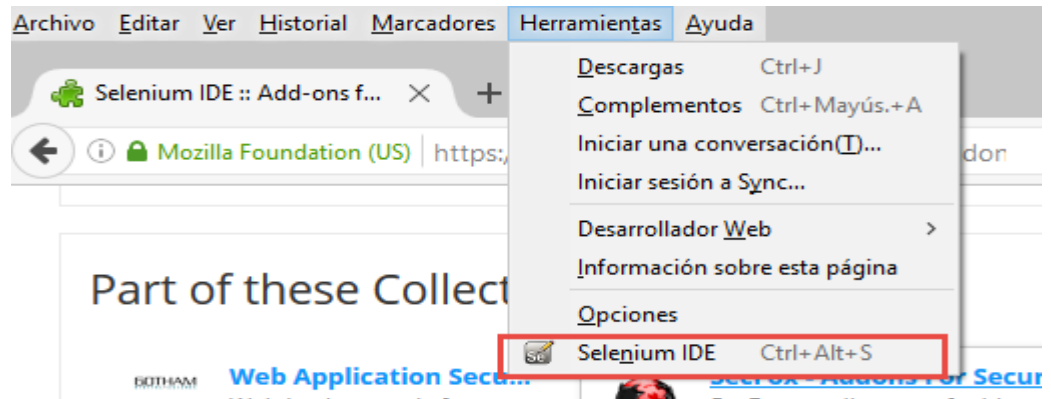


# Confirmar la instalación haciendo clic en el botón de Instalar

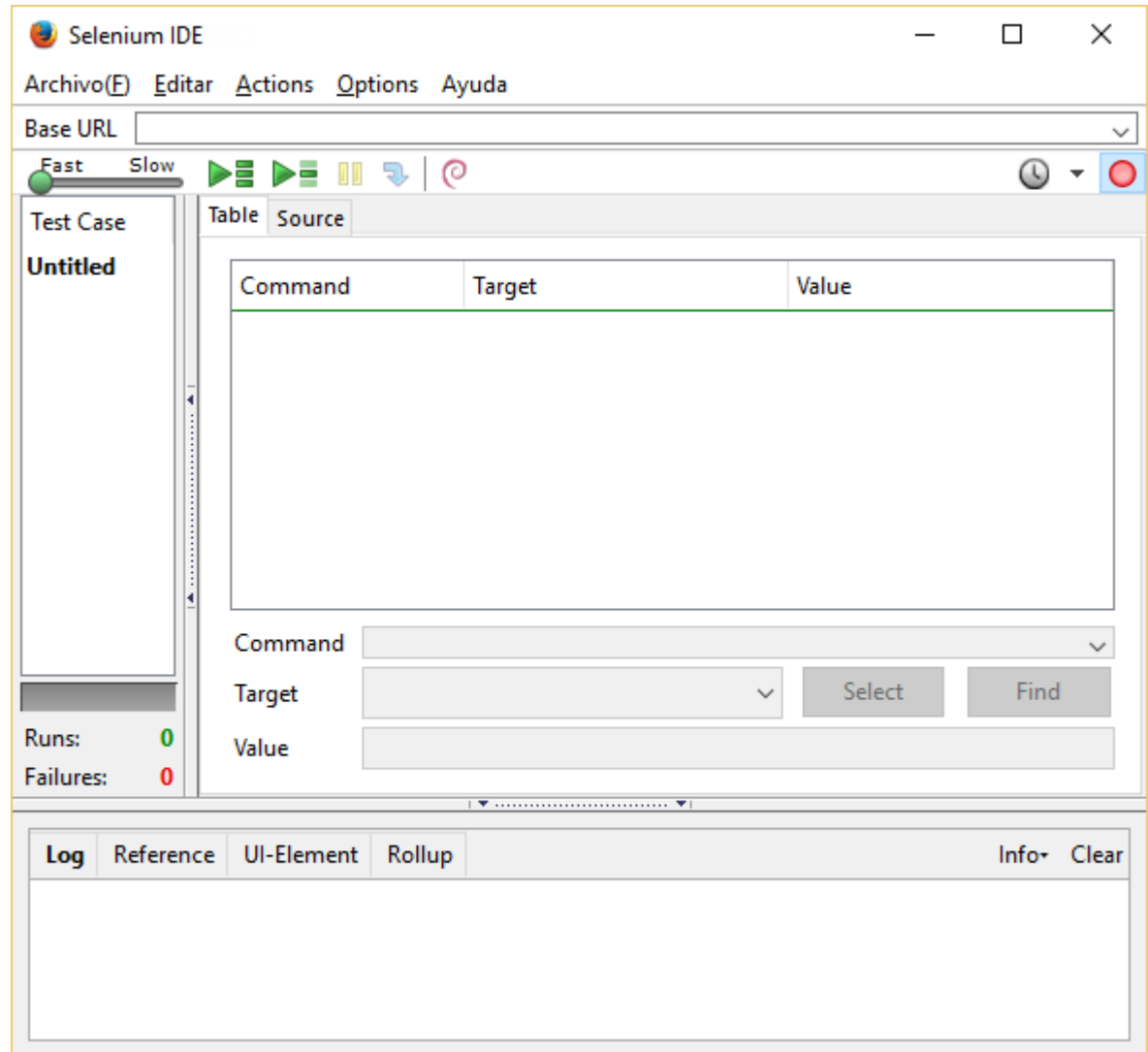


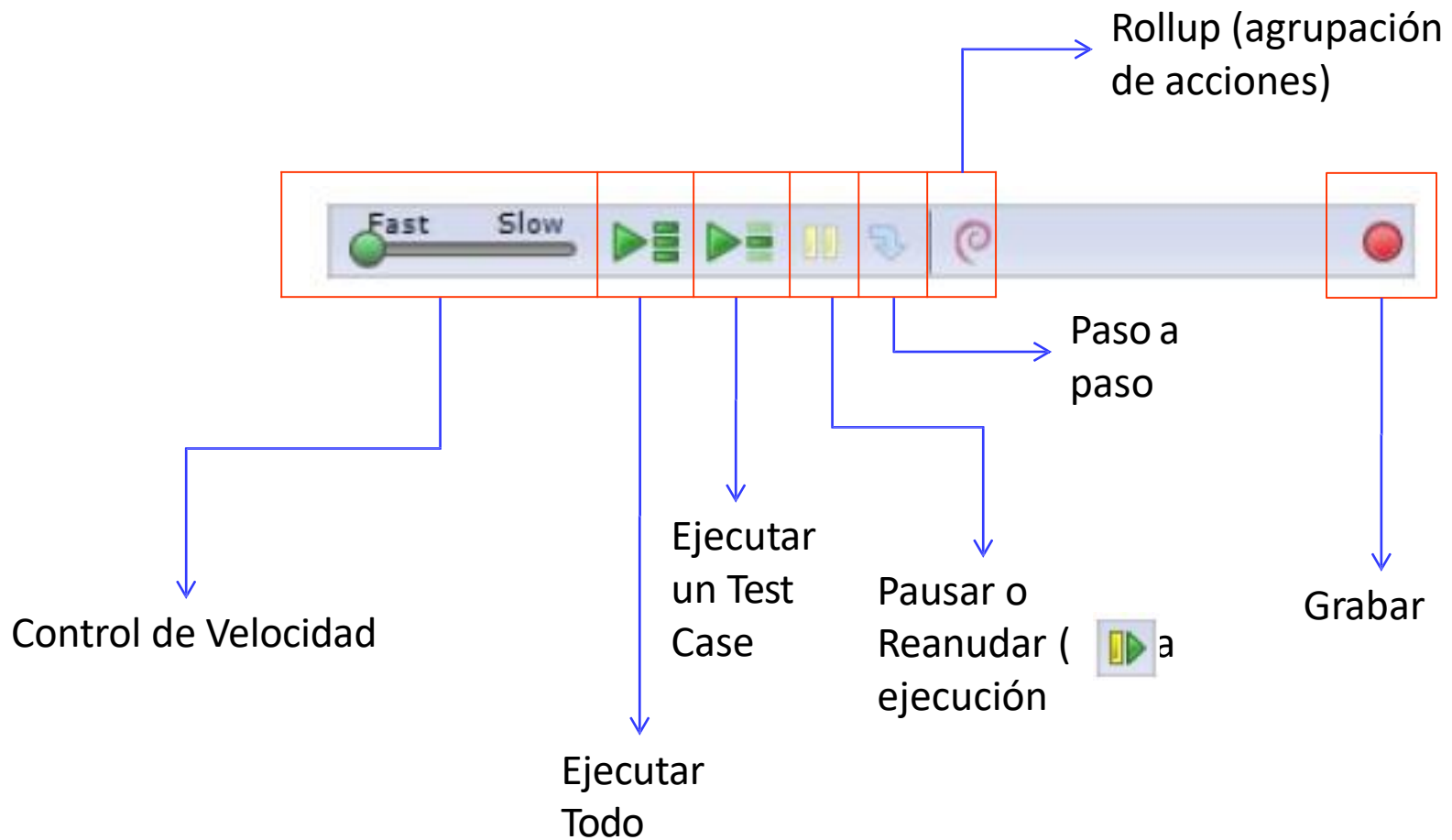
A través del  
menú

**Herramientas >  
Selenium IDE**



A través del  
menú  
**Herramientas >  
Selenium IDE**





Command	Target	Value
open	/	
waitForPageToLoad		
clickAndWait	xpath=id('menu_download')/a	
assertTitle	Downloads	
verifyText	xpath=id('mainContent')/h2	Downloads



Command	clickAndWait ▾	
Target	<input type="text" value="xpath=id('menu_download')/a"/>	<input type="button" value="Find"/>
Value	<input type="text"/>	

Log	Reference	UI-Element	Rollup	Info▼	Clear
[info]	Executing:	waitForPageToLoad			▲
[info]	Executing:	clickAndWait   xpath=id('menu_download')/a			
[info]	Executing:	assertTitle   Downloads			☰
[info]	Executing:	verifyText   xpath=id('mainContent')/h2   Downloads			▼

Log	Reference	UI-Element	Rollup
-----	-----------	------------	--------

### **clickAndWait(locator)**

Generated from **click(locator)**

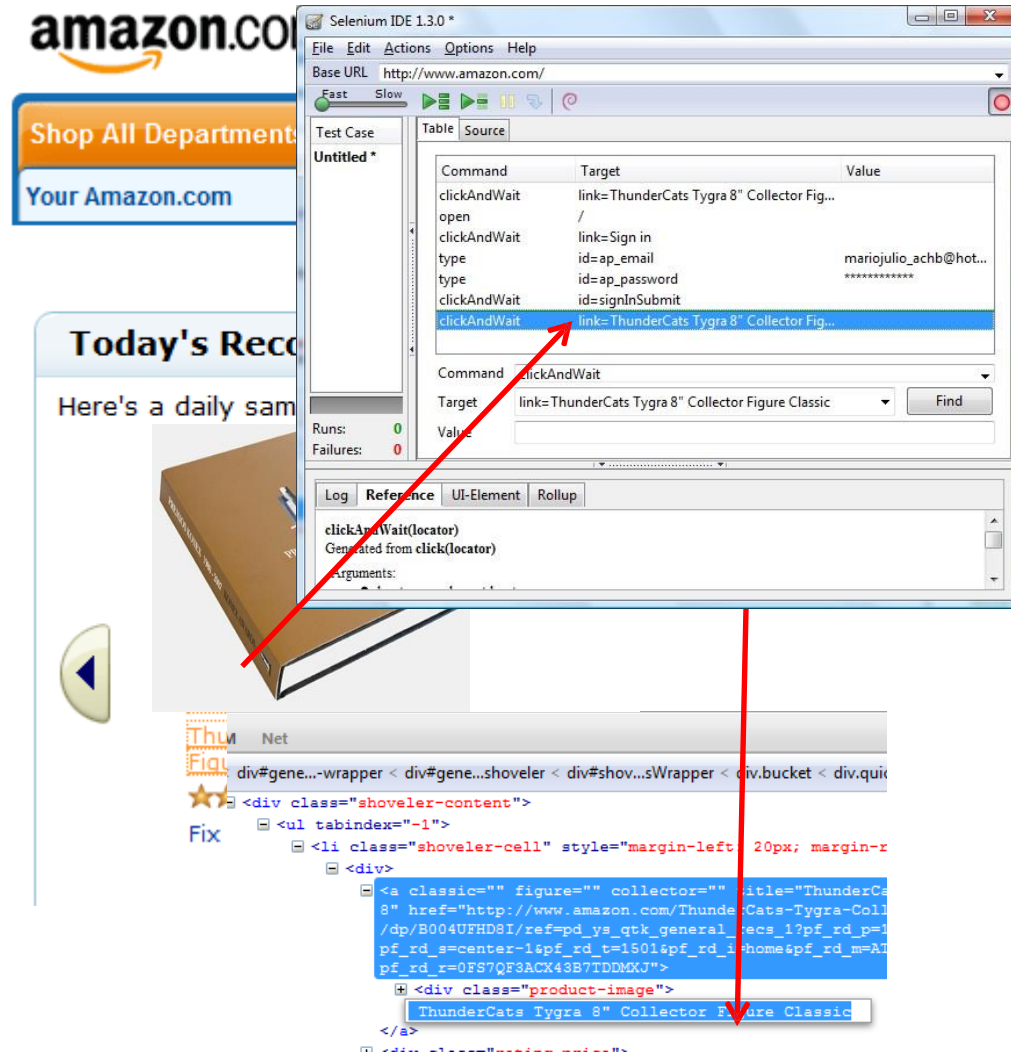
Arguments:

- locator - an element locator

Clicks on a link, button, checkbox or radio button. If the click action causes a new page to load (like a link usually does), call `waitForPageToLoad`.

- Inserción automática de comandos con acciones de usuario

- Links / Botones - ClickAndWait



The screenshot displays the Selenium IDE 1.3.0 interface. The top window shows the Amazon.com homepage with the 'Shop All Departments' button highlighted. The Selenium IDE window is open, showing a test case named 'Untitled \*'. The test case contains the following commands:

Command	Target	Value
clickAndWait	link=ThunderCats Tygra 8" Collector Fig...	
open	/	
clickAndWait	link=Sign in	
type	id=ap_email	mariojulio_achb@hotmail.com
type	id=ap_password	*****
clickAndWait	id=signInSubmit	
clickAndWait	link=ThunderCats Tygra 8" Collector Fig...	

The 'clickAndWait' command is highlighted in the table. A red arrow points from this command to the 'clickAndWait(locator)' command in the Log pane. The Log pane shows the command 'clickAndWait(locator)' generated from 'click(locator)' with arguments: 'link=ThunderCats Tygra 8" Collector Figure Classic'. Another red arrow points from the Log pane to the 'ThunderCats Tygra 8" Collector Figure Classic' link in the HTML source code.

The HTML source code snippet shows the following structure:

```
<div class="shovelers-content">  
  <ul tabindex="-1">  
    <li class="shovelers-cell" style="margin-left: 20px; margin-right: 20px; margin-bottom: 10px; margin-top: 10px;">  
      <div>  
        <a class="classic" figure="" collector="" title="ThunderCats Tygra 8" Collector Figure Classic" href="http://www.amazon.com/ThunderCats-Tygra-Coll.../dp/B004UFHD8I/ref=pd_vs_qtk_general_recs_1?pf_rd_p=1...pf_rd_s=center-1&pf_rd_t=1501&pf_rd_i=home&pf_rd_m=AT...pf_rd_r=0FS7QF3ACX43B7TDDMXJ">  
          <div class="product-image">  
            ThunderCats Tygra 8" Collector Figure Classic  
          </div>  
        </a>  
      </div>  
    </li>  
  </ul>  
</div>
```

- Inserción automática de comandos  
con acciones de usuario

- Textboxes /  
Text Areas - Click  
y Type

The image displays the Selenium IDE 1.3.0 interface with a test case named 'Untitled \*'. The test case contains the following commands:

Command	Target	Value
open	/	
clickAndWait	link=Sign in	
type	id=ap_email	mar...
type	id=ap_password	****
clickAndWait	id=signInSubmit	
click	id=twotabsearchtextbox	
type	id=twotabsearchtextbox	hola

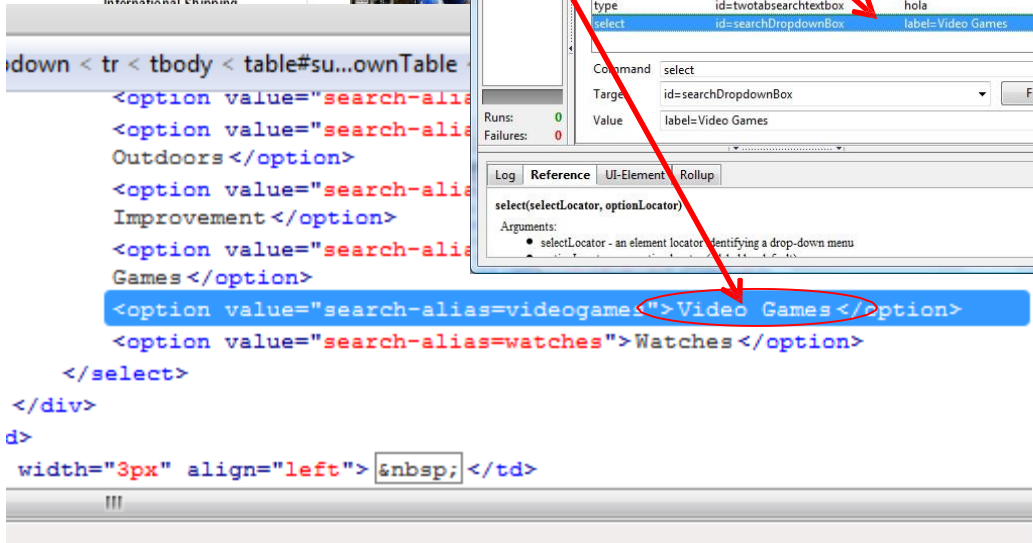
Below the table, the 'type' command is configured with Target 'id=twotabsearchtextbox' and Value 'hola'.

At the bottom, the 'Log' tab shows the command signature: `type(locator, value)` with the argument: `locator - an element locator`.

On the right, a snippet of the Amazon website's HTML is shown, with the search input field highlighted by a red circle and a red arrow pointing from the Selenium IDE interface:

```
<div class="nav-fill">  
  <div id="nav-search">  
    <div id="nav-bar-left"></div>  
    <div id="nav-bar-right"></div>  
    <div id="nav-searchbar">  
      <div id="nav-search-field">  
        <input type="text" id="twotabsearchtextbox" value="" name="field-keywords" autocomplete="off" />  
      </div>  
    </div>  
  </div>  
</div>
```

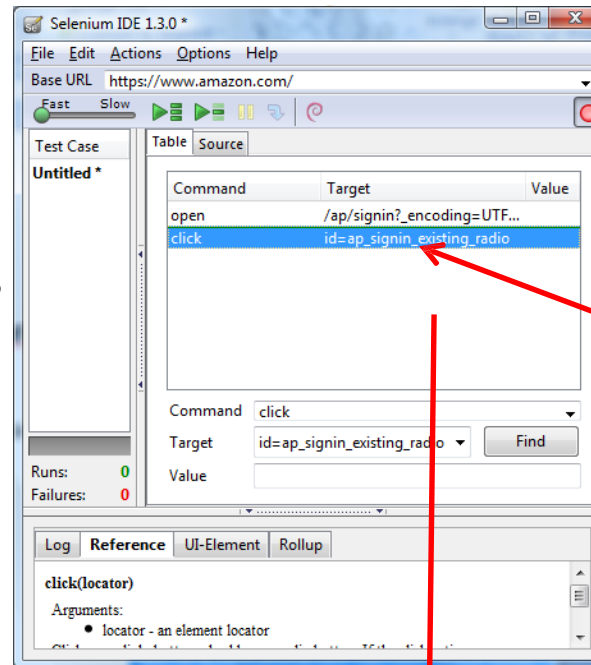
## - Dropdown - Select





-Inserción automática de comandos con acciones de usuario

- Checkboxes - Click



address?

s is:

amazon.com password?

- ☐ I am a new customer.  
(you'll create a password later)
- ☒ I am a returning customer,  
and my password is:

Sign in using our secure server

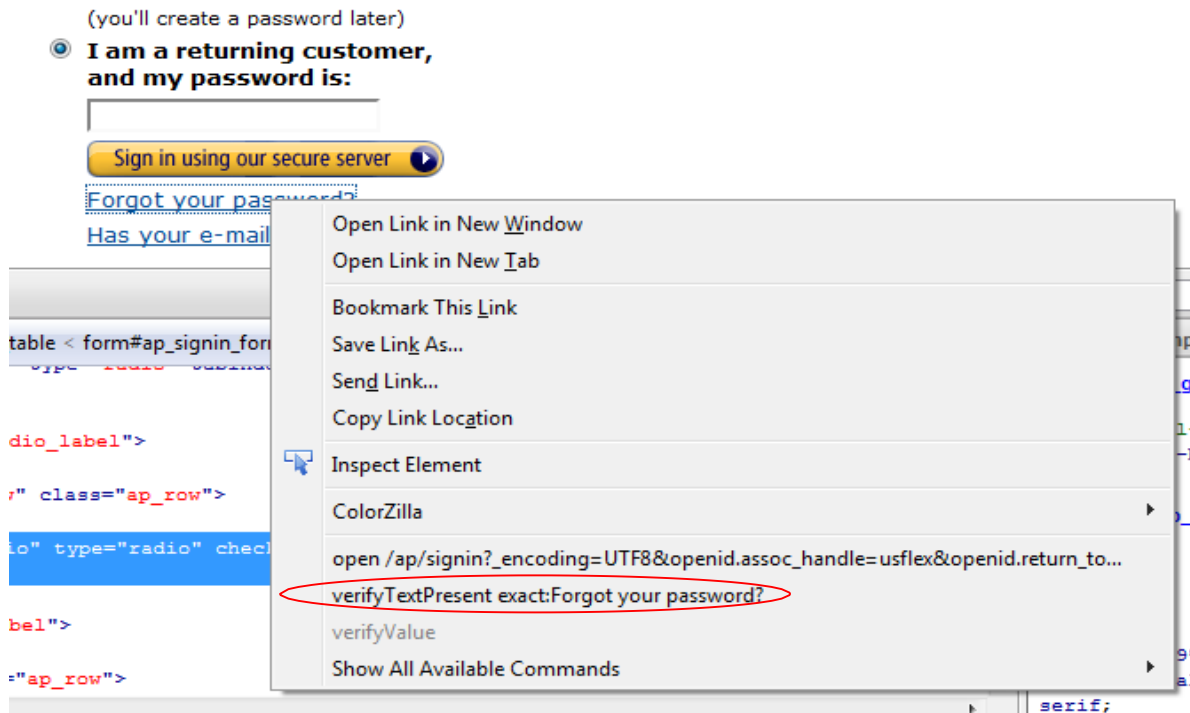
[Forgot your password?](#)

[Has your e-mail address change](#)

```
an.ap_col1 < div#ap_s...w.ap_row < div#ap_s...ap_table < form#ap_signin_form < div#signin-slot  
<input id="ap_signin_existing_radio" type="radio" checked="checked" value="true" />  
false); ">  
</span>  
<span id="" class="ap_col2 bold ap_radio_label">  
iv>  
v id="ap_signin1a_exist_cust_radio_row" class="ap_row">  
<span class="ap_coll1 ap_right">  
<input id="ap_signin_existing_radio" type="radio" checked="checked" value="true" />  
true); " name="create">  
</span>
```

# Grabación de

## Assertions / Verifications





# Assertions / Verifications



## -Assert VS Verify

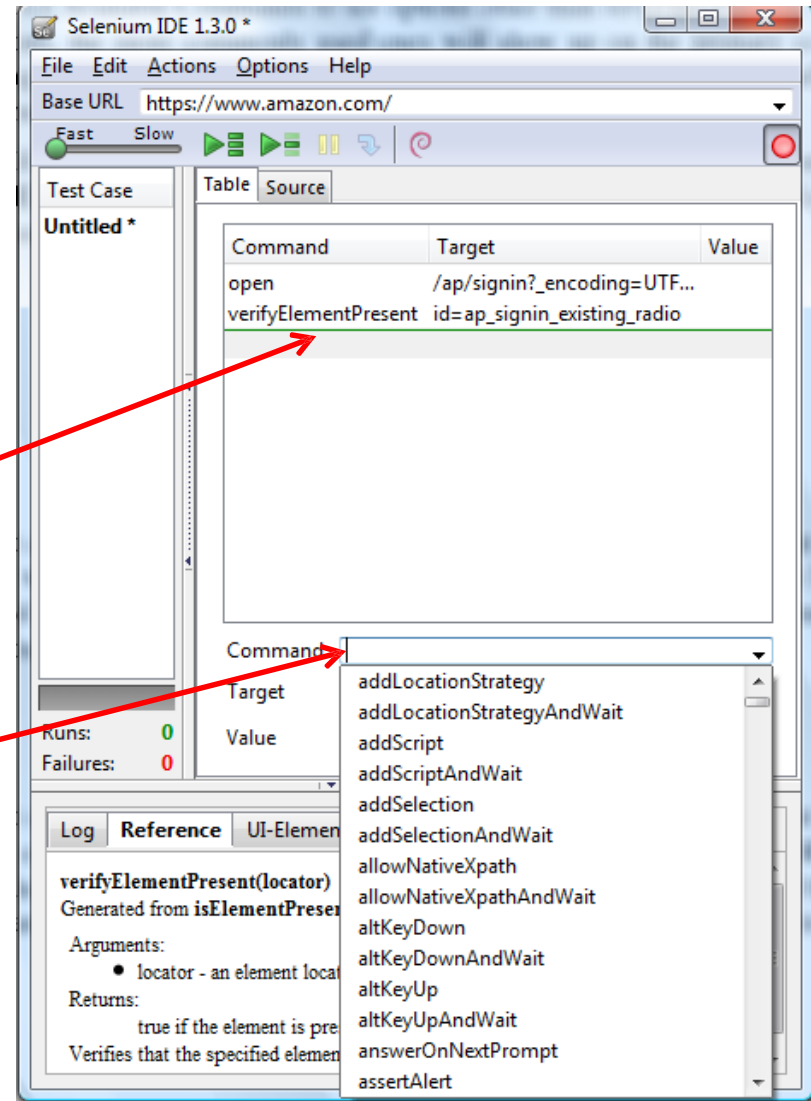
- La diferencia entre un Assert y un Verify está en que el primero finaliza el test case cuando el resultado no es el esperado mientras el último marca la inconsistencia como fallo y el test case continúa

## - Insertando Comandos

Vista de Tabla

1. Click aquí

2. Escribir aquí

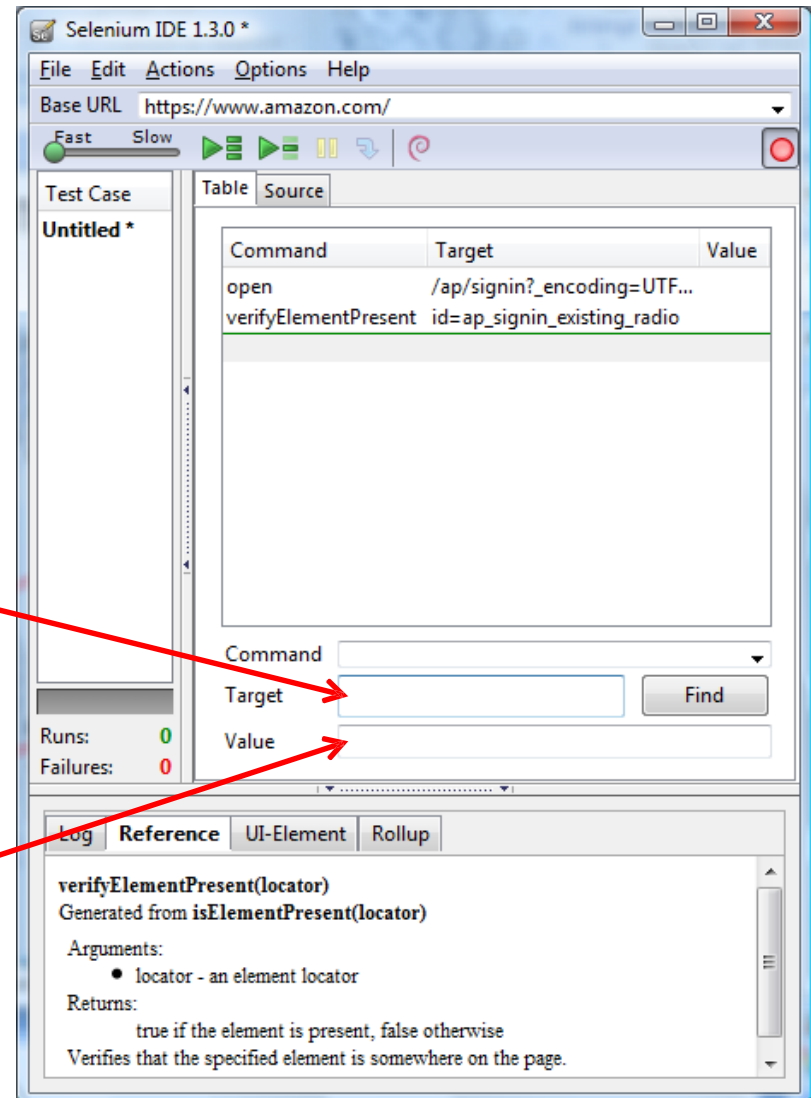


## - Insertando Comandos

Vista de Tabla

3. Especificar el elemento web

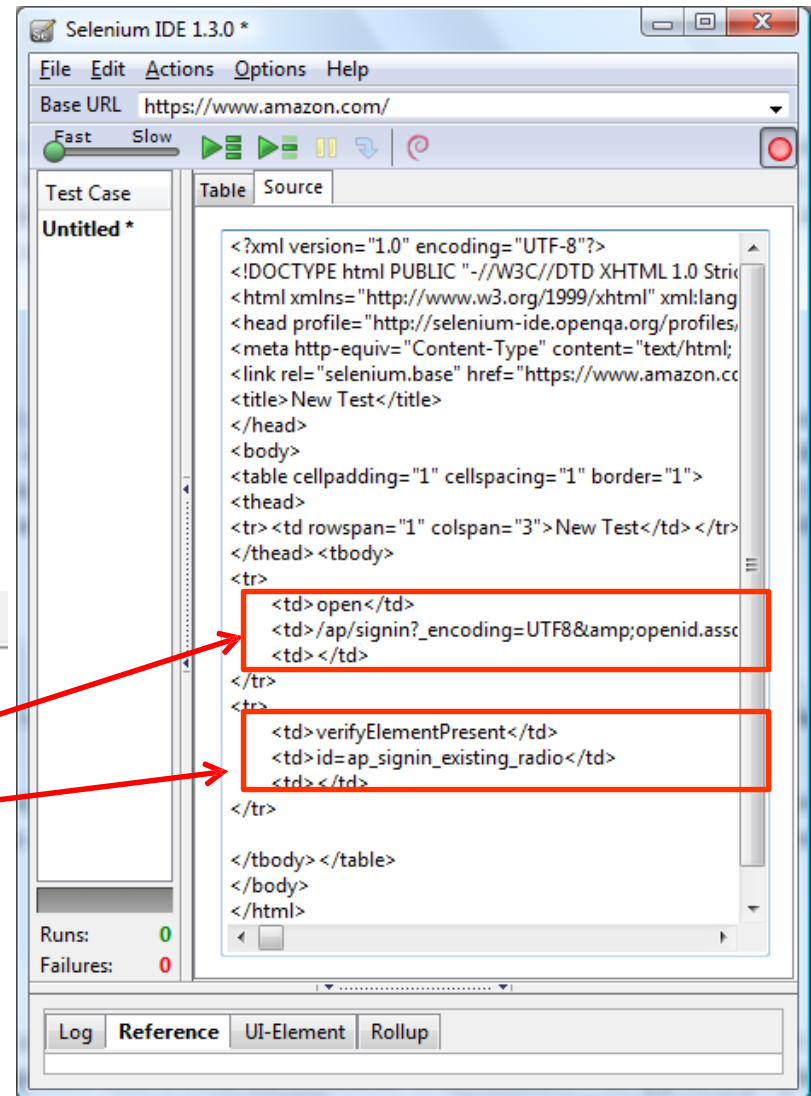
4. Especificar el valor atribuido a un elemento en caso aplique



## - Insertando Comandos

### Vista de Código Fuente

Table Source		
Command	Target	Value
open	/ap/signin?_encoding=UTF...	
verifyElementPresent	id=ap_signin_existing_radio	



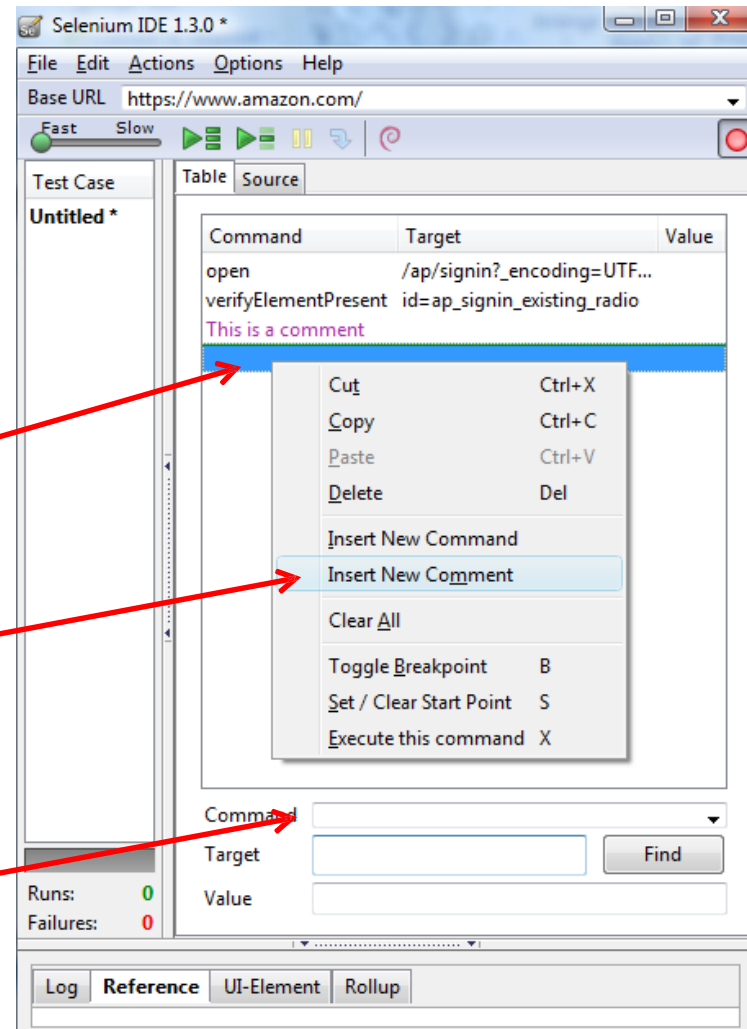
## - Insertando Comentarios

Vista de Tabla

1. Clic derecho

2. Seleccionar esta opción

3. Escribir el comentario

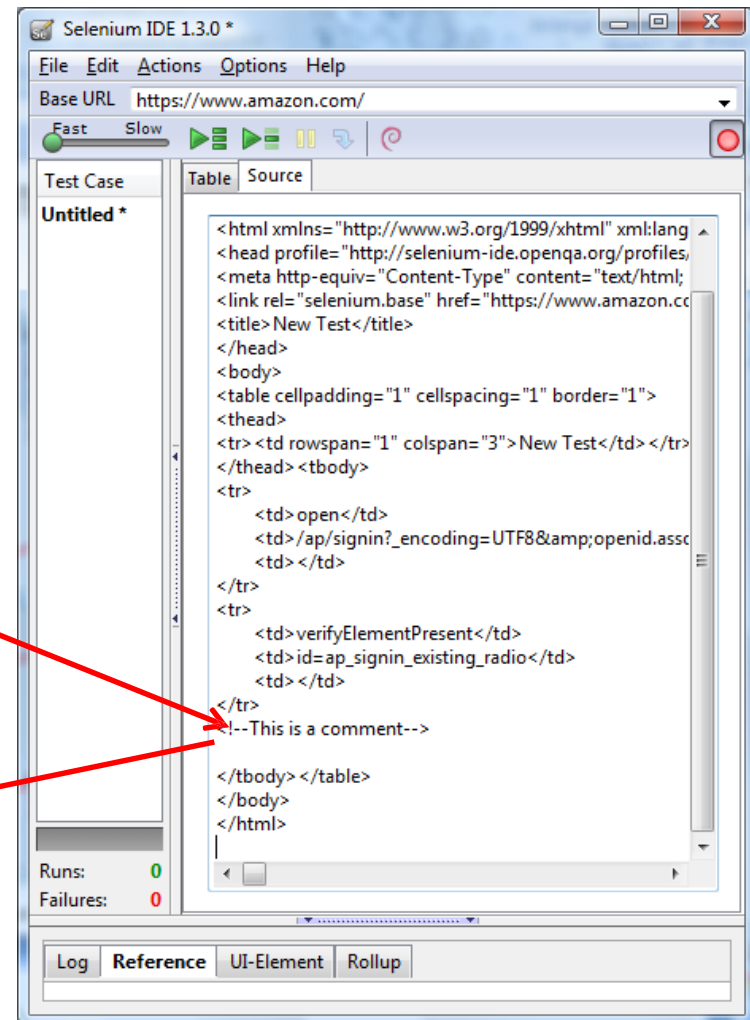


## - Insertando Comentarios

### Vista de Código Fuente

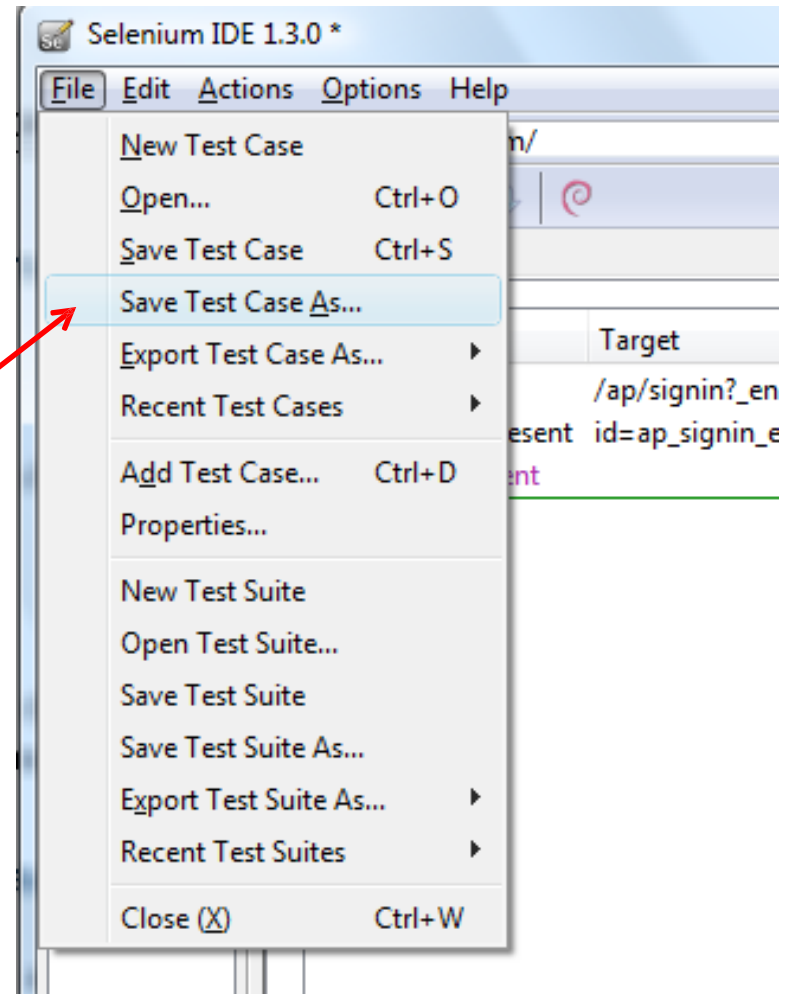
1. Insertar comentario como HTML

Command	Target	Value
open	/ap/signin?_encoding=UTF	
verifyElementPresent	id=ap_signin_existing_radio	
This is a comment		



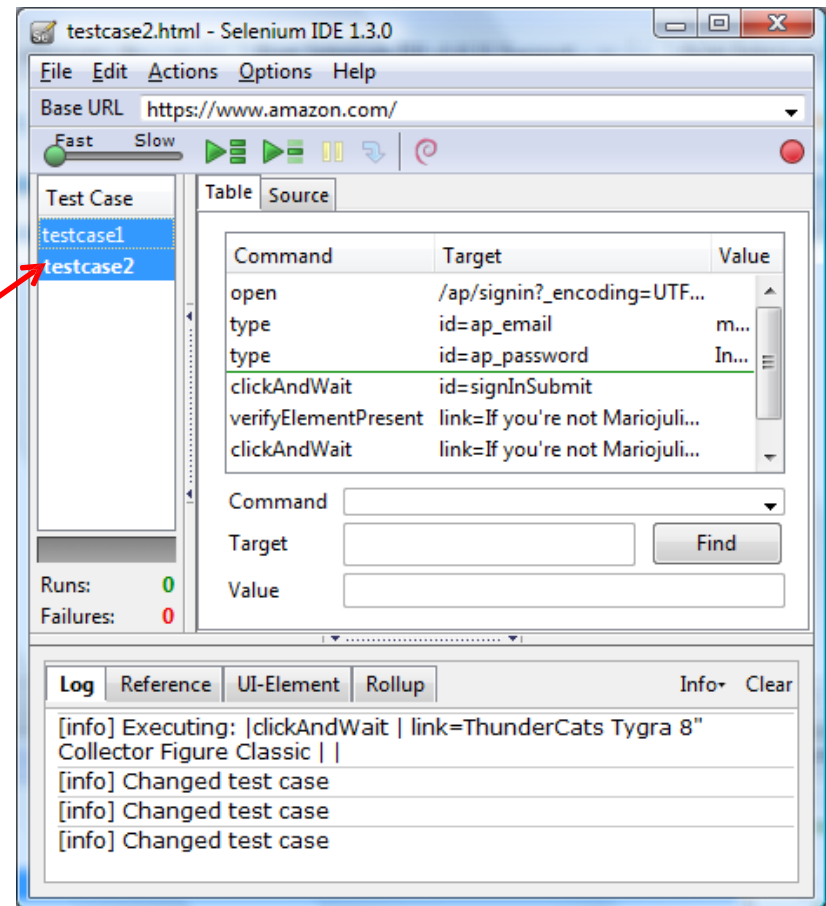
Desde el menú principal

1. Elegir esta  
opción



**Objetivo:** Evitar correr test cases uno por uno.

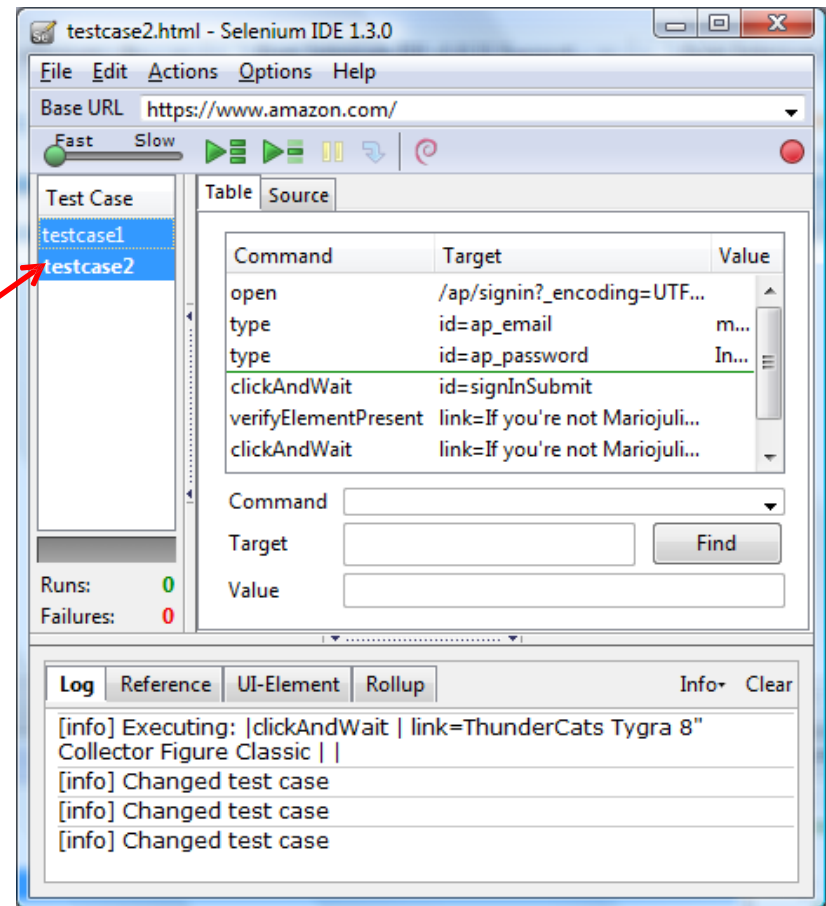
1. Más de dos test cases





**Objetivo:** Evitar correr test cases uno por uno.

1. Más de dos test cases



**Objetivo:** Evitar correr test cases uno por uno.

2. Elegir esta opción

