

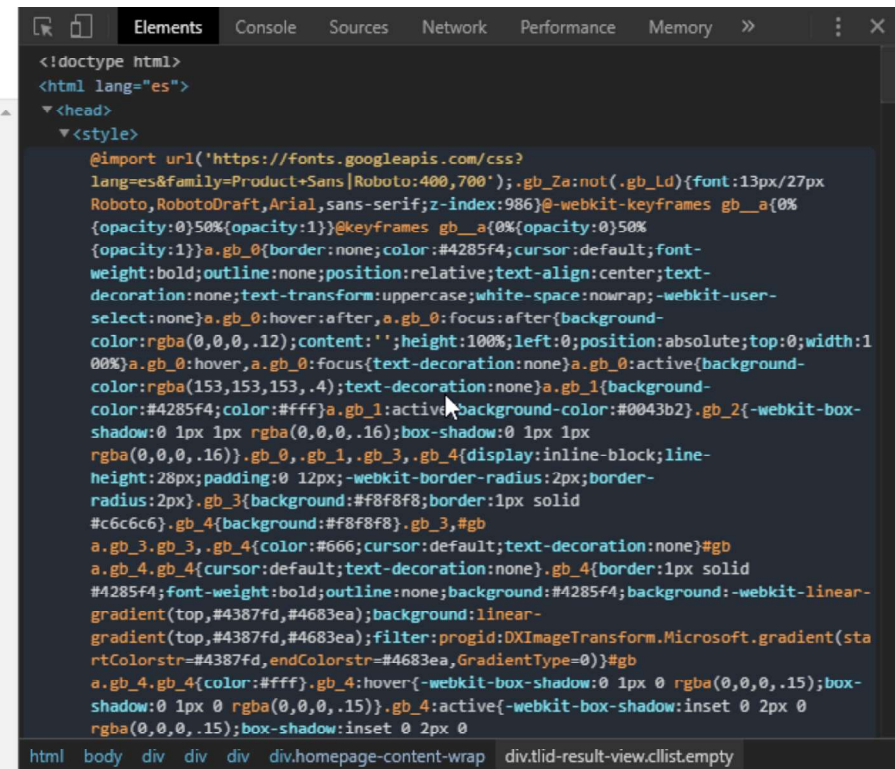
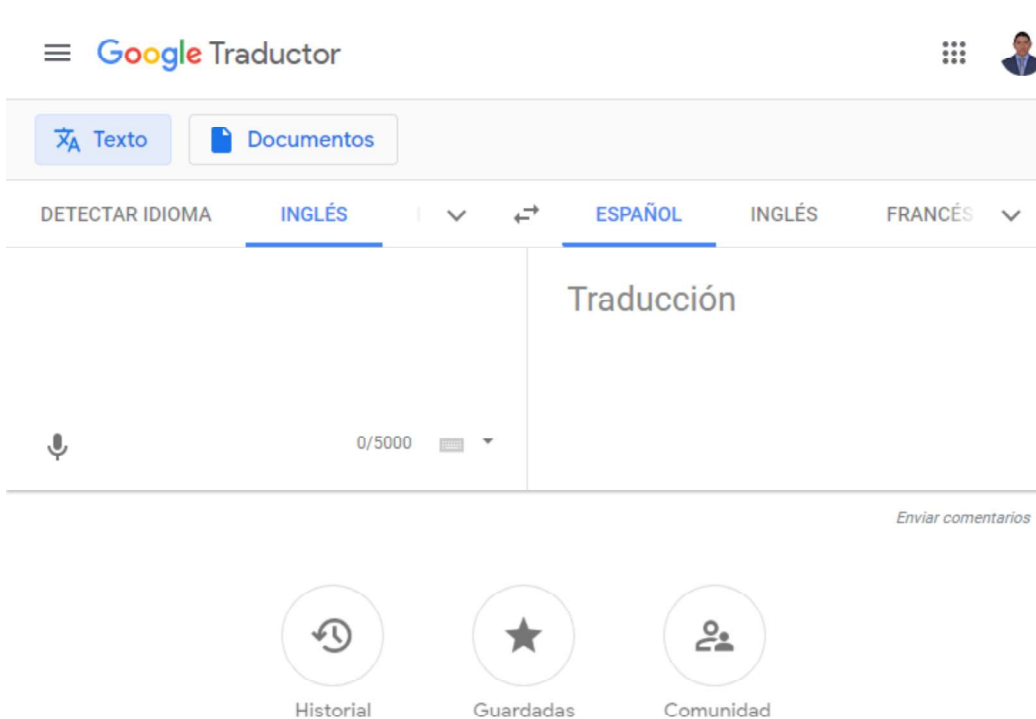
¿Qué son los localizadores(locators)?



¿Qué son los localizadores(locators)?

- Objetos que nos sirven para encontrar y hacer coincidir los elementos.
- Los localizadores son identificadores que nos permiten encontrar elementos en un pagina que se pueden usar nuestras pruebas.
- Atributos preestablecidos para buscar y encontrar elementos.

¿Qué son los localizadores(locators)?



Identificación de los elementos de la interfaz

Es muy importante que dicho localizador **se resuelva de forma única**, para que la prueba sea correcta, y la acción no se realice sobre un elemento indeseado. Por otra parte, también debe tenerse en cuenta que el identificador elegido **sea reutilizable en el futuro**.

Por ejemplo debe evitarse la elección de un identificador que cambie con cada nueva versión de la aplicación.

Por ello, es muy importante elegir de manera adecuada los identificadores que vamos a usar en cada caso.

Orden de los locators

- Localizadores: Nos permiten encontrar elementos (WebElement) en un pagina que se pueden usar nuestras pruebas. A continuación definiremos cada unos de los locators.
 - Id
 - Name
 - Class
 - TagName
 - LinkText
 - Selector de CSS
 - XPath

Id

El ID es un identificador para el elemento que normalmente asigna el desarrollador del código.

La identificación debe ser única en la página, pero los navegadores permiten la violación de esta regla, por lo que ocasionalmente no es única.

Este suele ser el mejor localizador donde solo hay uno de “esa cosa” en la página. Si aparece en una lista o tabla, entonces hay una buena posibilidad de que los ID puedan cambiar para futuras visitas a la página y se requiera un localizador más complejo.

```
▼ <label for="user_login">  
  "Username"  
  <br>  
  <input type="text" name="log" id="user_login" class=  
    "input" value size="20">  
</label>
```

Id

Es la mejor opción siempre y cuando la página HTML tenga definidos correctamente los identificadores de sus elementos.

La estrategia Id busca un elemento en la página que tenga un id atributo correspondiente al patrón especificado. `<label id="my_id" />` será emparejado por un localizador como `id=my_id` o simplemente `my_id`

PRO:

Se supone que cada ID es único, por lo que no hay posibilidad de que coincida con varios elementos.

<https://accounts.google.com/>

CONTRA:

Funciona bien solo en elementos con identificadores fijos y no generados.

<https://www.kayak.com.co/hotels>

Name

Los elementos de ubicación por nombre son muy similares a la ubicación por ID, excepto que usamos el prefijo "name =" en su lugar.

Cada campo de entrada en un formulario tiene un nombre. En general, los nombres de campo son únicos para el formulario, **pero no siempre**.

Un localizador de nombre de campo para campos de nombre de usuario y contraseña en una página de inicio de sesión suele funcionar muy bien. Pero cada botón de radio en un grupo relacionado tendrá el mismo nombre (y valores diferentes), por lo que se debe usar un localizador diferente para hacer clic en un botón específico en el grupo. En el ejemplo anterior, el campo de entrada podría ubicarse fácilmente utilizando la identificación o el nombre del campo.

Name

Por definición, el atributo 'name' de un elemento HTML no tiene que ser único, con lo que el uso de este método de localización no garantiza que la prueba se ejecute de la manera deseada

PROS :

Funciona bien con una lista fija de elementos similares.

<https://accounts.google.com/>

CONTRAS :

Difícil de usar con listas enlazadas a datos

<https://www.avianca.com/co/es/>

https://www.w3schools.com/tags/tryit.asp?filename=tryhtml5_input_type_radio

ClassName

Funciona muy parecido a name, aunque es mas inestable debido a que hoy por hoy los desarrolladores usan frameworks y sus clase se repiten mucho, por lo cual debes ser muy precavido y validar que el elemento sea único con la clase que buscas.

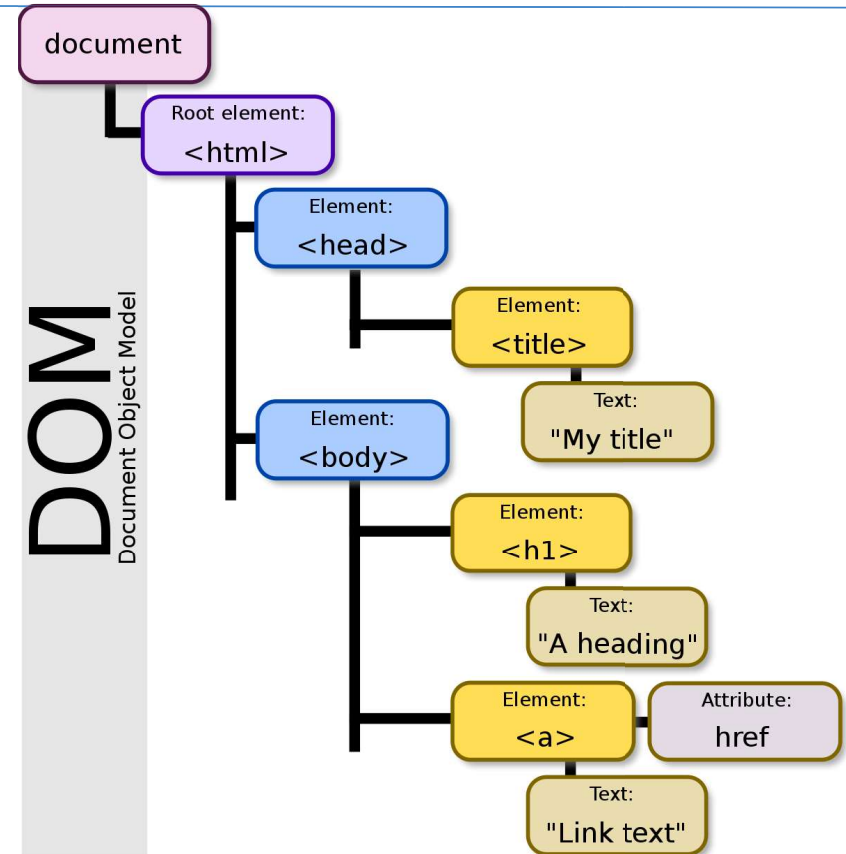
El localizador de nombre de clase proporciona el elemento que coincide con los valores especificados en el nombre de atributo "clase".

```
<td>  
<input id="email" class="inputtext" type="email" tabindex="1" value="" name="email">  
</td>
```

<https://www.airbnb.com.co/>

TagName

TagName es una parte de una estructura DOM donde cada elemento en una página se ha definido mediante una etiqueta como etiqueta de entrada, etiqueta de botón, etiqueta de anclaje, etc. Cada etiqueta tiene múltiples atributos como ID, nombre, clase de valor, etc. En cuanto a otros los localizadores en Selenium están preocupados, utilizamos estos valores de atributos de la etiqueta para localizar elementos. En el caso del tagName localizador en Selenium, simplemente usaremos el nombre de la etiqueta para identificar un elemento.



TagName

A continuación se muestra la estructura DOM de una página de inicio de sesión donde he resaltado los nombres de las etiquetas:

Campo de correo electrónico:

```
<input type="email" name="email" value="" placeholder="Email" required="required" autofocus="autofocus" class="form-control mt-3 form-control-lg">
```

Campo de contraseña:

```
<input type="password" name="password" placeholder="Password" class="form-control mt-3 form-control-lg" >
```

Botón de inicio de sesión:

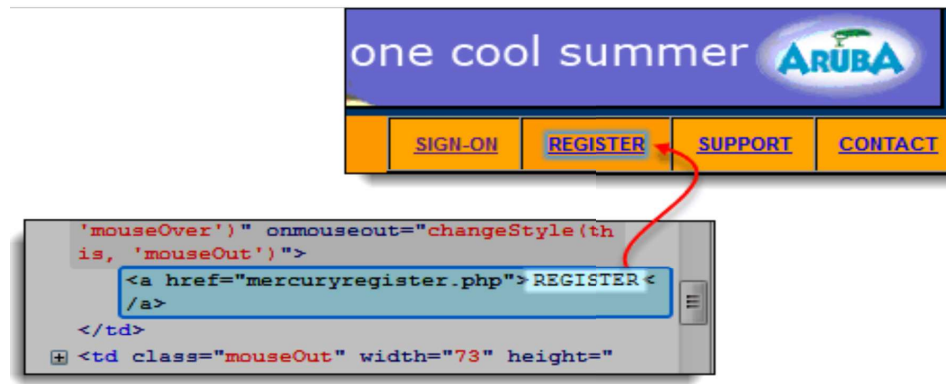
```
< button type="submit" class="btn btn-primary btn-lg btn-block mt-3">LOGIN< /button >
```

Olvidé el enlace de contraseña:

```
< button type="submit" class="btn btn-primary btn-lg btn-block mt-3">LOGIN< /button >
```

LinkText

Este tipo de localizador se aplica solo a textos de hipervínculo. Accedemos al enlace anteponiendo nuestro objetivo con "enlace =" y luego seguido por el texto del hipervínculo.



LinkText

Esta estrategia está destinada a seleccionar solo enlaces y selecciona el elemento de ancla que contiene el texto especificado: **link=The text of the link**

PROS :

- Si hay un elemento visible con texto que coincide exactamente con el texto proporcionado, el localizador de texto de enlace lo encontrará.
- Solo seleccionará elementos de anclaje
- Útil cuando se prueba la navegación

CONTRAS :

- Tienes que saber el texto del enlace antes

Selector de CSS

Los selectores CSS son patrones de cadena utilizados para identificar un elemento basado en una combinación de etiqueta HTML, id, clase y atributos. La localización mediante CSS Selector es más complicada que los métodos anteriores, pero es la estrategia de localización más común de los usuarios avanzados de Selenium porque puede acceder incluso a aquellos elementos que no tienen ID o nombre.

Selector de CSS

PROS :

Mucho más rápido que XPath

Ampliamente utilizado

Proporciona un buen equilibrio entre estructura y atributos.

Permite la selección de elementos por su contexto circundante.

CONTRAS :

Tienden a ser más complejos y requieren una curva de aprendizaje más pronunciada

Selector de CSS

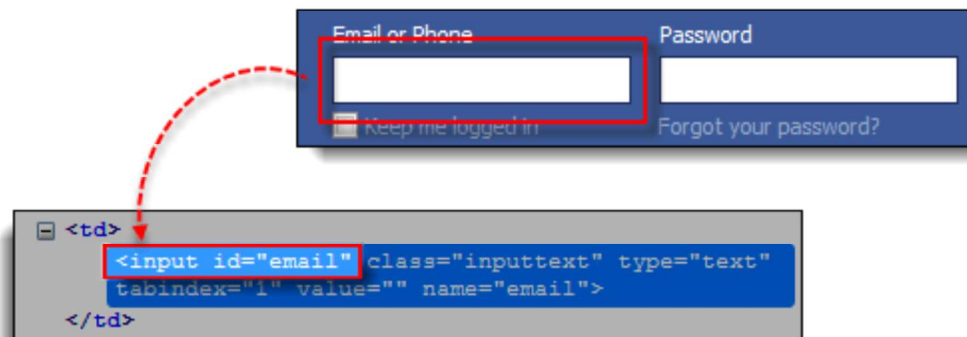
Los selectores CSS tienen muchos formatos, pero solo nos centraremos en los más comunes.

- Etiqueta e identificación
- Etiqueta y clase
- Etiqueta y atributo
- Etiqueta, clase y atributo
- Texto interno

Selector de CSS(Etiqueta e identificación)

Sintaxis	Descripción
css = etiqueta # id	<ul style="list-style-type: none"> Etiqueta = la etiqueta HTML del elemento al que se accede # = el signo hash. Esto siempre debe estar presente cuando se usa un selector CSS con ID id = la ID del elemento al que se accede

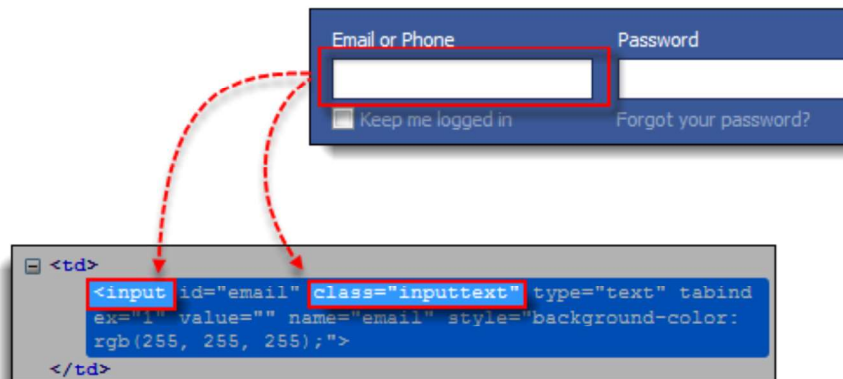
css = "input # email"



Selector de CSS(Etiqueta y clase)

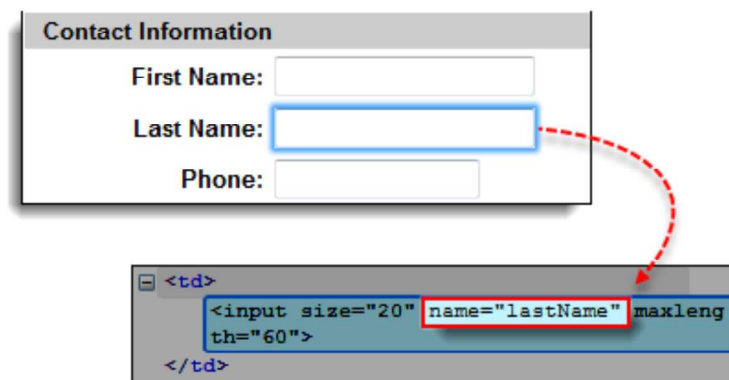
Sintaxis	Descripción
css = etiqueta . clase	<ul style="list-style-type: none"> •etiqueta = la etiqueta HTML del elemento al que se accede •. = el signo de punto. Esto siempre debe estar presente cuando se usa un selector de CSS con clase •clase = la clase del elemento al que se accede

css = "input.inputtext"



Selector de CSS(Etiqueta y atributo)

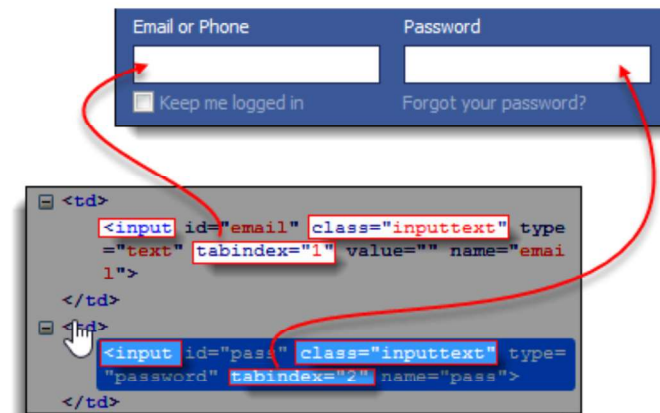
Sintaxis	Descripción
css = etiqueta [atributo = valor]	<ul style="list-style-type: none"> •etiqueta = la etiqueta HTML del elemento al que se accede •[y] = corchetes dentro de los cuales se colocará un atributo específico y su valor correspondiente •atributo = el atributo que se utilizará. Es aconsejable utilizar un atributo que sea exclusivo del elemento, como un nombre o ID. •valor = el valor correspondiente del atributo elegido.



css = “input [name = lastName]”

Selector de CSS(Etiqueta, clase y atributo)

Sintaxis	Descripción
css = tag.class [atributo = valor]	<ul style="list-style-type: none"> •etiqueta = la etiqueta HTML del elemento al que se accede •. = el signo de punto. Esto siempre debe estar presente cuando se usa un selector de CSS con clase •clase = la clase del elemento al que se accede •[y] = corchetes dentro de los cuales se colocará un atributo específico y su valor correspondiente •atributo = el atributo que se utilizará. Es aconsejable utilizar un atributo que sea exclusivo del elemento, como un nombre o ID. •valor = el valor correspondiente del atributo elegido.



css = "input.inputtext [tabindex = 1]"

Selector de CSS(Texto interno)

Sintaxis	Descripción
css = etiqueta : contiene (" texto interno ")	<ul style="list-style-type: none"> •etiqueta = la etiqueta HTML del elemento al que se accede •texto interno = el texto interno del elemento

```

<td align="right">
  <font size="2" face="Arial, Helvetica,
  sans-serif">Password:</font>
</td>
  
```

This is the inner text

User

Name:

Password:

css = font: contains ("Password:")
 Css=Font:contains("Name:")
 Css=Font:contains("User")

Xpath

Este método de identificación es similar al anterior, pero en este caso hace uso de la estructura XML que posee todo documento HTML, para así hacer referencia a los elementos mediante una ruta, ya sea absoluta (partiendo desde el elemento /) o relativa (partiendo de un elemento conocido).

Si bien DOM es el estándar reconocido para la navegación a través de un árbol de elementos HTML, XPath es la herramienta de navegación estándar para XML; y un documento HTML también es un documento XML (xHTML). XPath se usa en todas partes donde hay XML. Los localizadores válidos de XPath pueden ser:

`xpath=//button[@value="Blueberry"]`: coincide con el botón Blueberry
`//div[@id="pancakes"]/button[0]`: la misma cosa

Xpath

PROS :

Permite localizadores muy precisos
puede acceder a casi cualquier elemento, incluso aquellos sin atributos de clase, nombre o id.

CONTRAS :

Más lento que CSS

Se basa en la implementación de XPath del navegador que no siempre es completa (especialmente en IE) y, como tal, no se recomienda para las pruebas entre navegadores

Es el método más complicado para identificar elementos debido a demasiadas reglas y consideraciones diferentes.

Xpath

Recordemos: XPath está diseñado para permitir la navegación de documentos XML, con el propósito de seleccionar elementos individuales, atributos u otra parte de un documento XML para procesamiento específico. XPath produce localizadores confiables, pero en términos de rendimiento es más lento (especialmente en versiones anteriores de IE) en comparación con CSS Selector.

