cambie de ejemplo.com a google.com despues abrio una pagina de google y se cerro. Command line mostro:

C:\Users\MLimas\OneDrive - Epicor\Documents\QA Minds\Python>python mi\_script.py

DevTools listening on ws://127.0.0.1:59081/devtools/browser/46537f3a-fb21-4178-8c2d-2a8bd8f47d40

**Copilot said:**

**Copilot**

Parece que tu script de Python está intentando abrir una página web utilizando DevTools, pero la página se cierra inmediatamente. Esto podría deberse a varios factores. Aquí hay algunas cosas que podrías revisar:

1. **Revisa el código del script**: Asegúrate de que el script esté configurado correctamente para abrir y mantener la página abierta.
2. **Actualiza las dependencias**: Asegúrate de que todas las bibliotecas y dependencias estén actualizadas.
3. **Verifica la configuración de DevTools**: Asegúrate de que DevTools esté configurado correctamente y que no haya configuraciones que cierren la página automáticamente.
4. **Errores en el script**: Revisa si hay algún error en el script que pueda estar causando que la página se cierre.

Si puedes compartir el código del script, podría ayudarte a identificar el problema más específicamente. ¿Te gustaría intentar alguna de estas sugerencias?

Edit in Pages

**Copy**

AI-generated content may be incorrect

**You said:**

from selenium import webdriver

from selenium.webdriver.chrome.service import Service

from selenium.webdriver.chrome.options import Options

# Ruta al WebDriver descargado

driver\_path = 'C:\\Users\\MLimas\\OneDrive - Epicor\\Documents\\QA Minds\\chromedriver-win64\\chromedriver.exe'

service = Service(driver\_path)

# Configurar opciones de Chrome

chrome\_options = Options()

chrome\_options.add\_argument('--no-proxy-server')

chrome\_options.add\_argument('--disable-gpu')

driver = webdriver.Chrome(service=service, options=chrome\_options)

# Abre una página web

driver.get('https://www.google.com')

**Copilot said:**

**Copilot**

Gracias por compartir tu código. A primera vista, el script parece estar configurado correctamente para abrir una página web en Google Chrome. Sin embargo, el problema de que la página se cierre inmediatamente podría estar relacionado con varias cosas. Aquí hay algunas sugerencias para solucionar el problema:

1. **Agregar un tiempo de espera**: A veces, agregar un tiempo de espera puede ayudar a que la página se cargue completamente antes de que el script continúe. Puedes usar time.sleep() para esto.
2. **Manejo de excepciones**: Asegúrate de manejar cualquier excepción que pueda ocurrir durante la ejecución del script.

Aquí tienes una versión modificada de tu script con estas sugerencias:

Este script agrega un tiempo de espera de 10 segundos después de abrir la página y maneja cualquier excepción que pueda ocurrir. Si la página se cierra antes de que termine el tiempo de espera, el mensaje de error te dará más información sobre lo que podría estar fallando.

¿Te gustaría probar esta versión y ver si resuelve el problema?

Edit in Pages

**Copy**

AI-generated content may be incorrect

**You said:**

ok ya se abrio y se mantuvo abierta

**Copilot said:**

**Copilot**

¡Excelente! Me alegra saber que funcionó. 😊

Si tienes alguna otra pregunta o necesitas ayuda con algo más, ¡no dudes en decírmelo! ¿Hay algo más en tu script o algún otro proyecto en el que estés trabajando?

Edit in Pages

**Copy**

AI-generated content may be incorrect

**You said:**

tengo que automatizar una pagina con selenium. ya instale python y chromedriver, por eso utilice ese codigo. Que mas tengo que hacer para tener configurado selenium correctamente?

**Copilot said:**

**Copilot**

¡Perfecto! Ya tienes Python y ChromeDriver instalados, así que estás en buen camino. Aquí hay algunos pasos adicionales para asegurarte de que Selenium esté configurado correctamente y puedas automatizar una página web sin problemas:

1. **Instalar Selenium**: Asegúrate de tener Selenium instalado en tu entorno de Python. Puedes hacerlo ejecutando el siguiente comando:
2. **Actualizar ChromeDriver**: Asegúrate de que la versión de ChromeDriver coincida con la versión de tu navegador Chrome. Puedes verificar la versión de Chrome y descargar la versión correspondiente de ChromeDriver desde aquí.
3. **Configurar el entorno**: Asegúrate de que el entorno de Python esté configurado correctamente y que las variables de entorno estén establecidas para que Python y ChromeDriver se puedan encontrar.
4. **Escribir y probar scripts**: Comienza escribiendo scripts sencillos para automatizar tareas básicas en la página web. Aquí tienes un ejemplo de cómo interactuar con elementos en una página web:
5. **Documentación y recursos**: Consulta la documentación oficial de Selenium para obtener más detalles sobre cómo interactuar con diferentes elementos en una página web y cómo manejar eventos y excepciones.
6. **Práctica y experimentación**: La mejor manera de aprender es practicando. Intenta automatizar diferentes tareas y experimenta con diferentes métodos y opciones disponibles en Selenium.

¿Hay alguna funcionalidad específica que te gustaría automatizar en la página web? ¡Estoy aquí para ayudarte!

Edit in Pages

**Copy**

AI-generated content may be incorrect

**You said:**

como puedo revisar si ya tengo instalado selenium hay alguna manera de revisarlo antes de instalarlo?

**Copilot said:**

**Copilot**

Sí, puedes verificar si Selenium ya está instalado en tu entorno de Python utilizando el siguiente comando en la línea de comandos:

Este comando mostrará información sobre el paquete Selenium si está instalado. Si no está instalado, no mostrará nada. Aquí tienes un ejemplo de lo que podrías ver si Selenium está instalado:

Si no ves esta información, puedes proceder a instalar Selenium con el comando:

¿Te gustaría intentar verificar si Selenium ya está instalado en tu entorno?

Edit in Pages

**Copy**

AI-generated content may be incorrect

**You said:**

vi esto:

Name: selenium

Version: 4.30.0

Summary: Official Python bindings for Selenium WebDriver

Home-page: https://www.selenium.dev

Author:

Author-email:

License: Apache 2.0

Location: C:\Users\MLimas\AppData\Local\Programs\Python\Python313\Lib\site-packages

Requires: certifi, trio, trio-websocket, typing\_extensions, urllib3, websocket-client

Required-by:

**Copilot said:**

**Copilot**

¡Perfecto! Eso significa que Selenium ya está instalado en tu entorno de Python. 🎉

Ahora que tienes Selenium instalado, puedes continuar escribiendo y ejecutando tus scripts de automatización. Si necesitas ayuda con alguna funcionalidad específica o tienes alguna pregunta sobre cómo interactuar con elementos en una página web, ¡no dudes en preguntar!

¿Hay alguna tarea específica que te gustaría automatizar en la página web? Estoy aquí para ayudarte a lograrlo.

Edit in Pages

**Copy**

AI-generated content may be incorrect

**You said:**

si quiero que me ayudes paso a paso a automatizar una pagina de pruebas gratis, tienes algunas ideas de paginas?

**Copilot said:**

**Copilot**

¡Claro! Aquí tienes algunas páginas web gratuitas que son ideales para practicar la automatización de pruebas con Selenium:

1. **DemoQA**: Ofrece una variedad de elementos típicos de sitios web, como formularios, ventanas emergentes, y listas ordenables. Es excelente para practicar diferentes interacciones de automatización

1

.

* + [**DemoQA**](https://demoqa.com/)

1. **The-Internet**: Proporcionado por Elemental Selenium, este sitio incluye diversas situaciones de prueba, como formularios de inicio de sesión, arrastrar y soltar, y más

1

.

* + The-Internet

1. **Swag Labs**: Un sitio de demostración de comercio electrónico donde puedes practicar la navegación, selección y compra de productos, y gestión del carrito de compras.
   * [**Swag Labs**](https://www.saucedemo.com/)
2. **Automation Exercise**: Ofrece ejercicios prácticos y desafiantes para mejorar tus habilidades en la automatización de pruebas

2

.

* + [**Automation Exercise**](https://automationexercise.com/)

1. **Selenium Easy**: Similar a los anteriores, con muchos ejemplos concretos y micro aplicaciones para practicar

1

.

* + Selenium Easy

Podemos empezar con **DemoQA**. Aquí tienes un ejemplo de cómo podrías automatizar una tarea en este sitio:

Este script abre la página de DemoQA, espera a que se cargue, y luego hace clic en el botón de "Elements". Puedes modificar y expandir este script para interactuar con otros elementos en la página.

¿Te gustaría probar este ejemplo o necesitas ayuda con alguna otra funcionalidad específica? ¡Estoy aquí para ayudarte!

**1**Favicon typeSitios de prueba para practicar - Federico Toledo**2**Favicon typeSitios web para practicar automatización de pruebas WebUIFavicon typeSitios de prueba para practicar - Federico Toledo

Show more

Edit in Pages

**Copy**

AI-generated content may be incorrect

**You said:**

quiero automatizar dos scenarios: https://demoqa.com/automation-practice-form

1. llenar la forma de manera correcta y guardar.

2. Llenar algun campo incorrectamente y guardar salga algun errror

Puedes ayudarme?

tambien guiame paso a paso para hacerlo desde visual studio code. PReguntame antes de avanzar al siguiente paso

**Copilot said:**

**Copilot**

¡Claro! Vamos a automatizar los dos escenarios en la página de práctica de formularios de DemoQA. Te guiaré paso a paso desde Visual Studio Code. Empecemos con la configuración inicial y luego avanzaremos a los escenarios específicos.

**Paso 1: Configuración Inicial**

1. **Instalar Visual Studio Code**: Si aún no lo tienes instalado, puedes descargarlo desde aquí.
2. **Instalar Python**: Asegúrate de tener Python instalado. Puedes descargarlo desde aquí.
3. **Instalar Selenium**: Abre Visual Studio Code y abre una terminal (puedes hacerlo desde Terminal > New Terminal). Luego, ejecuta el siguiente comando para instalar Selenium:
4. **Descargar ChromeDriver**: Asegúrate de tener ChromeDriver descargado y que la versión coincida con tu navegador Chrome. Puedes descargarlo desde aquí.

**Paso 2: Crear el Script de Automatización**

1. **Crear un nuevo archivo**: En Visual Studio Code, crea un nuevo archivo llamado automation\_script.py.
2. **Escribir el código inicial**: Copia y pega el siguiente código en tu archivo automation\_script.py. Este código abrirá la página de práctica de formularios de DemoQA.

**Paso 3: Ejecutar el Script**

1. **Ejecutar el script**: En Visual Studio Code, abre la terminal y navega hasta el directorio donde guardaste automation\_script.py. Luego, ejecuta el script con el siguiente comando:

Esto debería abrir la página de práctica de formularios en tu navegador.

¿Te gustaría proceder con estos pasos iniciales y luego avanzar a los escenarios específicos de automatización? ¡Avísame cuando estés listo para continuar!

Edit in Pages

**Copy**

AI-generated content may be incorrect

**You said:**

ya se abrio la pagina. Ahora sigamos con el escenario de llenar todos los campos del formulario con valores validos.

**Copilot said:**

**Copilot**

¡Perfecto! Vamos a automatizar el llenado del formulario con valores válidos. Aquí tienes el código paso a paso para llenar todos los campos del formulario y enviarlo:

**Paso 4: Llenar el Formulario con Valores Válidos**

1. **Actualizar el script**: Añade el siguiente código a tu archivo automation\_script.py para llenar el formulario con valores válidos.

**Paso 5: Ejecutar el Script**

1. **Ejecutar el script**: En Visual Studio Code, abre la terminal y navega hasta el directorio donde guardaste automation\_script.py. Luego, ejecuta el script con el siguiente comando:

Este script llenará todos los campos del formulario con valores válidos y lo enviará.

¿Te gustaría probar este ejemplo y luego proceder al escenario de llenar algún campo incorrectamente? ¡Avísame cuando estés listo para continuar!

Edit in Pages

**Copy**

AI-generated content may be incorrect

**You said:**

como sabre si pasa o no pasa el script? mostraria algun mensaje o algo que me avise?

**Copilot said:**

**Copilot**

¡Buena pregunta! Para saber si el script ha pasado o no, podemos agregar una verificación después de enviar el formulario para comprobar si aparece un mensaje de confirmación o algún elemento específico que indique que el formulario se ha enviado correctamente.

**Paso 6: Verificar el Resultado del Script**

1. **Actualizar el script**: Añade el siguiente código al final de tu script para verificar si el formulario se ha enviado correctamente.

**Paso 7: Ejecutar el Script**

1. **Ejecutar el script**: En Visual Studio Code, abre la terminal y navega hasta el directorio donde guardaste automation\_script.py. Luego, ejecuta el script con el siguiente comando:

Este script llenará todos los campos del formulario con valores válidos, lo enviará y luego verificará si aparece un mensaje de confirmación. Si el formulario se ha enviado correctamente, verás el mensaje "El formulario se ha enviado correctamente." en la terminal.

¿Te gustaría probar este ejemplo y luego proceder al escenario de llenar algún campo incorrectamente? ¡Avísame cuando estés listo para continuar!

Edit in Pages

**Copy**

AI-generated content may be incorrect

**You said:**

vi que se abrio la pagina, lleno la info y la cerro. despues en cmd ocurrio esto:

C:\Users\MLimas\OneDrive - Epicor\Documents\QA Minds\Selenium>python automation\_script.py

DevTools listening on ws://127.0.0.1:60862/devtools/browser/177e3cbe-6c4b-4aee-b70e-3843202b61e4

Ocurrió un error: Message: element click intercepted: Element <label title="" for="gender-radio-1" class="custom-control-label">...</label> is not clickable at point (400, 241). Other element would receive the click: <iframe frameborder="0" src="https://020f6d43706a0f28e5071fb09e30097c.safeframe.googlesyndication.com/safeframe/1-0-41/html/container.html" id="google\_ads\_iframe\_/21849154601,22343295815/Ad.Plus-Anchor\_0" title="3rd party ad content" name="" scrolling="no" marginwidth="0" marginheight="0" width="728" height="90" data-is-safeframe="true" sandbox="allow-forms allow-popups allow-popups-to-escape-sandbox allow-same-origin allow-scripts allow-top-navigation-by-user-activation" allow="private-state-token-redemption;attribution-reporting" aria-label="Advertisement" tabindex="0" data-google-container-id="1" style="border: 0px; vertical-align: bottom;" data-load-complete="true"></iframe>

(Session info: chrome=135.0.7049.42)

Stacktrace:

GetHandleVerifier [0x00007FF6E26B1F55+78133]

GetHandleVerifier [0x00007FF6E26B1FB0+78224]

(No symbol) [0x00007FF6E24791BA]

(No symbol) [0x00007FF6E24D70D9]

(No symbol) [0x00007FF6E24D4A92]

(No symbol) [0x00007FF6E24D1B31]

(No symbol) [0x00007FF6E24D0A31]

(No symbol) [0x00007FF6E24C2164]

(No symbol) [0x00007FF6E24F715A]

(No symbol) [0x00007FF6E24C1A16]

(No symbol) [0x00007FF6E24F7370]

(No symbol) [0x00007FF6E251F21F]

(No symbol) [0x00007FF6E24F6F33]

(No symbol) [0x00007FF6E24C0358]

(No symbol) [0x00007FF6E24C10C3]

GetHandleVerifier [0x00007FF6E297BA8D+3001453]

GetHandleVerifier [0x00007FF6E2975E72+2977874]

GetHandleVerifier [0x00007FF6E299497D+3103581]

GetHandleVerifier [0x00007FF6E26CC7EA+186826]

GetHandleVerifier [0x00007FF6E26D43FF+218591]

GetHandleVerifier [0x00007FF6E26B9D94+110452]

GetHandleVerifier [0x00007FF6E26B9F42+110882]

GetHandleVerifier [0x00007FF6E26A0379+5465]

BaseThreadInitThunk [0x00007FF807FCE8D7+23]

RtlUserThreadStart [0x00007FF80953BF6C+44]

**Copilot said:**

**Copilot**

El error "element click intercepted" ocurre cuando otro elemento está bloqueando el elemento que intentas hacer clic. En este caso, parece que un anuncio está bloqueando el botón de género. Aquí hay algunas soluciones para resolver este problema:

**Paso 8: Resolver el Error "Element Click Intercepted"**

1. **Esperar a que el elemento sea clickeable**: Utiliza WebDriverWait para esperar a que el elemento sea clickeable antes de intentar hacer clic.
2. **Desplazar la página**: Desplaza la página para asegurarte de que el elemento esté visible y no bloqueado por otros elementos.
3. **Cerrar el anuncio**: Si el anuncio está bloqueando el elemento, intenta cerrar el anuncio antes de hacer clic en el elemento.

Aquí tienes una versión modificada del script que incluye estas soluciones:

**Paso 9: Ejecutar el Script**

1. **Ejecutar el script**: En Visual Studio Code, abre la terminal y navega hasta el directorio donde guardaste automation\_script.py. Luego, ejecuta el script con el siguiente comando:

Este script debería resolver el problema de "element click intercepted" y verificar si el formulario se ha enviado correctamente.

¿Te gustaría probar este ejemplo y luego proceder al escenario de llenar algún campo incorrectamente? ¡Avísame cuando estés listo para continuar!

**1**Favicon typeelement click intercepted Error in selenium - Stack Overflow**2**Favicon typeorg.openqa.selenium.ElementClickInterceptedException: element click ...Favicon typeHow To Resolve Element Click Intercepted Exception in Selenium? - QASource

Show more

Edit in Pages

**Copy**

AI-generated content may be incorrect

**You said:**

ahora veo esto en cmd:

C:\Users\MLimas\OneDrive - Epicor\Documents\QA Minds\Selenium>python automation\_script.py

DevTools listening on ws://127.0.0.1:61476/devtools/browser/fa239f91-e06e-4b3b-b513-b6cc957d11ee

Ocurrió un error: Message: element click intercepted: Element <label title="" for="hobbies-checkbox-1" class="custom-control-label">...</label> is not clickable at point (418, 310). Other element would receive the click: <iframe frameborder="0" src="https://61073965aa12941f2ff618b8328cacd1.safeframe.googlesyndication.com/safeframe/1-0-41/html/container.html" id="google\_ads\_iframe\_/21849154601,22343295815/Ad.Plus-Anchor\_0" title="3rd party ad content" name="" scrolling="no" marginwidth="0" marginheight="0" width="728" height="90" data-is-safeframe="true" sandbox="allow-forms allow-popups allow-popups-to-escape-sandbox allow-same-origin allow-scripts allow-top-navigation-by-user-activation" allow="private-state-token-redemption;attribution-reporting" aria-label="Advertisement" tabindex="0" data-google-container-id="1" style="border: 0px; vertical-align: bottom;" data-load-complete="true"></iframe>

(Session info: chrome=135.0.7049.42)

Stacktrace:

GetHandleVerifier [0x00007FF6E26B1F55+78133]

GetHandleVerifier [0x00007FF6E26B1FB0+78224]

(No symbol) [0x00007FF6E24791BA]

(No symbol) [0x00007FF6E24D70D9]

(No symbol) [0x00007FF6E24D4A92]

(No symbol) [0x00007FF6E24D1B31]

(No symbol) [0x00007FF6E24D0A31]

(No symbol) [0x00007FF6E24C2164]

(No symbol) [0x00007FF6E24F715A]

(No symbol) [0x00007FF6E24C1A16]

(No symbol) [0x00007FF6E24F7370]

(No symbol) [0x00007FF6E251F21F]

(No symbol) [0x00007FF6E24F6F33]

(No symbol) [0x00007FF6E24C0358]

(No symbol) [0x00007FF6E24C10C3]

GetHandleVerifier [0x00007FF6E297BA8D+3001453]

GetHandleVerifier [0x00007FF6E2975E72+2977874]

GetHandleVerifier [0x00007FF6E299497D+3103581]

GetHandleVerifier [0x00007FF6E26CC7EA+186826]

GetHandleVerifier [0x00007FF6E26D43FF+218591]

GetHandleVerifier [0x00007FF6E26B9D94+110452]

GetHandleVerifier [0x00007FF6E26B9F42+110882]

GetHandleVerifier [0x00007FF6E26A0379+5465]

BaseThreadInitThunk [0x00007FF807FCE8D7+23]

RtlUserThreadStart [0x00007FF80953BF6C+44]

**Copilot said:**

**Copilot**

Parece que el problema persiste debido a que un anuncio está bloqueando el elemento que intentas hacer clic. Vamos a intentar cerrar el anuncio antes de interactuar con los elementos del formulario. Aquí tienes una versión modificada del script que incluye el cierre del anuncio:

**Paso 10: Cerrar el Anuncio y Continuar con la Automatización**

1. **Actualizar el script**: Añade el siguiente código a tu archivo automation\_script.py para cerrar el anuncio antes de interactuar con los elementos del formulario.

**Paso 11: Ejecutar el Script**

1. **Ejecutar el script**: En Visual Studio Code, abre la terminal y navega hasta el directorio donde guardaste automation\_script.py. Luego, ejecuta el script con el siguiente comando:

Este script intentará cerrar el anuncio antes de interactuar con los elementos del formulario. Si el anuncio no está presente o no se puede cerrar, el script continuará con la automatización del formulario.

¿Te gustaría probar este ejemplo y luego proceder al escenario de llenar algún campo incorrectamente? ¡Avísame cuando estés listo para continuar!

Favicon typeelement click intercepted Error in selenium - Stack OverflowFavicon typeorg.openqa.selenium.ElementClickInterceptedException: element click ...Favicon typeHow To Resolve Element Click Intercepted Exception in Selenium? - QASource

Show more

Edit in Pages

**Copy**

AI-generated content may be incorrect

**You said:**

C:\Users\MLimas\OneDrive - Epicor\Documents\QA Minds\Selenium>python automation\_script.py

DevTools listening on ws://127.0.0.1:61727/devtools/browser/3408cc6d-6e97-4546-9bb0-62c2148b2f3b

No se encontró el anuncio o no se pudo cerrar.

Ocurrió un error: Message: no such element: Unable to locate element: {"method":"css selector","selector":"[id="firstName"]"}

(Session info: chrome=135.0.7049.42); For documentation on this error, please visit: https://www.selenium.dev/documentation/webdriver/troubleshooting/errors#no-such-element-exception

Stacktrace:

GetHandleVerifier [0x00007FF6E26B1F55+78133]

GetHandleVerifier [0x00007FF6E26B1FB0+78224]

(No symbol) [0x00007FF6E24791BA]

(No symbol) [0x00007FF6E24CF19D]

(No symbol) [0x00007FF6E24CF44C]

(No symbol) [0x00007FF6E25223D7]

(No symbol) [0x00007FF6E24F719F]

(No symbol) [0x00007FF6E251F21F]

(No symbol) [0x00007FF6E24F6F33]

(No symbol) [0x00007FF6E24C0358]

(No symbol) [0x00007FF6E24C10C3]

GetHandleVerifier [0x00007FF6E297BA8D+3001453]

GetHandleVerifier [0x00007FF6E2975E72+2977874]

GetHandleVerifier [0x00007FF6E299497D+3103581]

GetHandleVerifier [0x00007FF6E26CC7EA+186826]

GetHandleVerifier [0x00007FF6E26D43FF+218591]

GetHandleVerifier [0x00007FF6E26B9D94+110452]

GetHandleVerifier [0x00007FF6E26B9F42+110882]

GetHandleVerifier [0x00007FF6E26A0379+5465]

BaseThreadInitThunk [0x00007FF807FCE8D7+23]

RtlUserThreadStart [0x00007FF80953BF6C+44]

**Copilot said:**

**Copilot**

El error "no such element: Unable to locate element" ocurre cuando Selenium no puede encontrar el elemento especificado en la página. Esto puede deberse a varios factores, como que el elemento no esté presente en el DOM en el momento de la búsqueda o que el selector utilizado no sea correcto.

**Paso 12: Solucionar el Error "No Such Element"**

1. **Verificar el selector**: Asegúrate de que el selector utilizado para encontrar el elemento sea correcto. Puedes inspeccionar el elemento en el navegador y verificar el ID, clase, o cualquier otro atributo utilizado en el selector.
2. **Esperar a que el elemento esté presente**: Utiliza WebDriverWait para esperar a que el elemento esté presente en el DOM antes de intentar interactuar con él.
3. **Desplazar la página**: Asegúrate de que el elemento esté visible en la ventana del navegador.

Aquí tienes una versión modificada del script que incluye estas soluciones:

**Paso 13: Ejecutar el Script**

1. **Ejecutar el script**: En Visual Studio Code, abre la terminal y navega hasta el directorio donde guardaste automation\_script.py. Luego, ejecuta el script con el siguiente comando:

Este script debería resolver el problema de "no such element" y verificar si el formulario se ha enviado correctamente.

¿Te gustaría probar este ejemplo y luego proceder al escenario de llenar algún campo incorrectamente? ¡Avísame cuando estés listo para continuar!

**1**Favicon typeWhat is the best way to avoid NoSuchElementException in Selenium?**2**Favicon typeorg.openqa.selenium.NoSuchElementException: no such elementFavicon typeNo Such Element Exception in Selenium: How Do You Handle It ...

Show more

Edit in Pages

**Copy**

AI-generated content may be incorrect

**You said:**

tendremos muchos problemas con los anuncios: utilicemos esta pagina: https://www.saucedemo.com/

En el login utilicemos el siguiente user: standard\_user password:secret\_sauce

despues demos click en el boton de login

Hay que incluir una validacion si nos redirige a este site: https://www.saucedemo.com/inventory.html, la prueba falla. de lo contraria falla. Puedes generarme un mensaje o algo para saber que la prueba paso?

**Copilot said:**

**Copilot**

¡Entendido! Vamos a utilizar la página de **Swag Labs** para realizar la automatización del login. Aquí tienes el script paso a paso para automatizar el login y validar la redirección:

**Paso 1: Configuración Inicial**

1. **Actualizar el script**: Crea un nuevo archivo llamado saucedemo\_login.py en Visual Studio Code y copia el siguiente código:

**Paso 2: Ejecutar el Script**

1. **Ejecutar el script**: En Visual Studio Code, abre la terminal y navega hasta el directorio donde guardaste saucedemo\_login.py. Luego, ejecuta el script con el siguiente comando:

Este script realizará el login en la página de Swag Labs con el usuario standard\_user y la contraseña secret\_sauce, y luego verificará si la redirección a la página de inventario (https://www.saucedemo.com/inventory.html) fue exitosa. Si la redirección es correcta, verás el mensaje "La prueba pasó: Redirigido correctamente a la página de inventario." en la terminal. De lo contrario, verás el mensaje "La prueba falló: No se redirigió a la página de inventario."

¿Te gustaría probar este ejemplo y ver si funciona correctamente? ¡Avísame si necesitas más ayuda o si estás listo para continuar con otro escenario!

Edit in Pages

**Copy**

AI-generated content may be incorrect

**You said:**

C:\Users\MLimas\OneDrive - Epicor\Documents\QA Minds\Selenium>python saucedemo\_login.py

DevTools listening on ws://127.0.0.1:51268/devtools/browser/db298cc8-0dd1-43fa-9735-6c004037c1ee

La prueba pasó: Redirigido correctamente a la página de inventario.

**Copilot said:**

**Copilot**

¡Excelente! 🎉 Me alegra saber que la prueba pasó y que la redirección a la página de inventario fue exitosa.

Ahora que hemos completado el primer escenario, podemos proceder al segundo escenario: llenar algún campo incorrectamente y verificar que aparezca un error. ¿Te gustaría continuar con este escenario? Aquí tienes el plan:

**Paso 3: Automatizar el Escenario de Error**

1. **Actualizar el script**: Modifica el script para ingresar un usuario incorrecto y verificar que aparezca un mensaje de error.

**Paso 4: Ejecutar el Script**

1. **Ejecutar el script**: En Visual Studio Code, abre la terminal y navega hasta el directorio donde guardaste saucedemo\_login.py. Luego, ejecuta el script con el siguiente comando:

Este script ingresará un usuario incorrecto y verificará si aparece un mensaje de error. Si el mensaje de error aparece, verás el mensaje "La prueba pasó: Apareció un mensaje de error." en la terminal. De lo contrario, verás el mensaje "La prueba falló: No apareció un mensaje de error."

¿Te gustaría probar este ejemplo y ver si funciona correctamente? ¡Avísame si necesitas más ayuda o si estás listo para continuar con otro escenario!

Edit in Pages

**Copy**

AI-generated content may be incorrect

**You said:**

excelente hagamos una segunda prueba. ahora donde ingresamos un usuario y password invalido. Por ejemplo:

Usuario: Cleo!@#

**Copilot said:**

**Copilot**

¡Perfecto! Vamos a realizar una segunda prueba donde ingresamos un usuario y contraseña inválidos. Aquí tienes el script modificado para este escenario:

**Paso 5: Automatizar el Escenario de Usuario y Contraseña Inválidos**

1. **Actualizar el script**: Modifica el script para ingresar un usuario y contraseña inválidos y verificar que aparezca un mensaje de error.

**Paso 6: Ejecutar el Script**

1. **Ejecutar el script**: En Visual Studio Code, abre la terminal y navega hasta el directorio donde guardaste saucedemo\_login.py. Luego, ejecuta el script con el siguiente comando:

Este script ingresará un usuario y contraseña inválidos y verificará si aparece un mensaje de error. Si el mensaje de error aparece, verás el mensaje "La prueba pasó: Apareció un mensaje de error." en la terminal. De lo contrario, verás el mensaje "La prueba falló: No apareció un mensaje de error."

¿Te gustaría probar este ejemplo y ver si funciona correctamente? ¡Avísame si necesitas más ayuda o si estás listo para continuar con otro escenario!

Edit in Pages

**Copy**

AI-generated content may be incorrect

**You said:**

excelente hagamos una segunda prueba. ahora donde ingresamos un usuario y password invalido. Por ejemplo:

Usuario: Cleo!@#

Password: BadPassword

hay que validar que alado de usuario y password se agregan unos iconos rojos con tacha de error.

Abajo de password se despliega un error. que mensiona : Epic sadface: Username and password do not match any user in this service

crees poder apoyarme?

**Copilot said:**

**Copilot**

¡Claro que sí! Vamos a realizar una prueba donde ingresamos un usuario y contraseña inválidos, y verificamos que aparezcan los íconos rojos de error junto a los campos de usuario y contraseña, así como el mensaje de error debajo del campo de contraseña.

**Paso 7: Automatizar el Escenario de Usuario y Contraseña Inválidos con Validación de Errores**

1. **Actualizar el script**: Modifica el script para ingresar un usuario y contraseña inválidos, y verificar la presencia de los íconos rojos de error y el mensaje de error.

**Paso 8: Ejecutar el Script**

1. **Ejecutar el script**: En Visual Studio Code, abre la terminal y navega hasta el directorio donde guardaste saucedemo\_login.py. Luego, ejecuta el script con el siguiente comando:

Este script ingresará un usuario y contraseña inválidos, verificará la presencia de los íconos rojos de error junto a los campos de usuario y contraseña, y comprobará que el mensaje de error debajo del campo de contraseña sea "Epic sadface: Username and password do not match any user in this service". Si todas las verificaciones son correctas, verás los mensajes correspondientes en la terminal.

¿Te gustaría probar este ejemplo y ver si funciona correctamente? ¡Avísame si necesitas más ayuda o si estás listo para continuar con otro escenario!

Edit in Pages

**Copy**

AI-generated content may be incorrect

**You said:**

me ocurrio el siguiente error:

C:\Users\MLimas\OneDrive - Epicor\Documents\QA Minds\Selenium>python Incorrect\_User\_Password.py

DevTools listening on ws://127.0.0.1:51965/devtools/browser/e34a2f4d-0197-43e5-8e19-afc0d630d734

La prueba pasó: Apareció un mensaje de error.

Ocurrió un error: Message: no such element: Unable to locate element: {"method":"xpath","selector":"//input[@id="user-name"]/following-sibling::svg"}

(Session info: chrome=135.0.7049.42); For documentation on this error, please visit: https://www.selenium.dev/documentation/webdriver/troubleshooting/errors#no-such-element-exception

Stacktrace:

GetHandleVerifier [0x00007FF6E26B1F55+78133]

GetHandleVerifier [0x00007FF6E26B1FB0+78224]

(No symbol) [0x00007FF6E24791BA]

(No symbol) [0x00007FF6E24CF19D]

(No symbol) [0x00007FF6E24CF44C]

(No symbol) [0x00007FF6E25223D7]

(No symbol) [0x00007FF6E24F719F]

(No symbol) [0x00007FF6E251F21F]

(No symbol) [0x00007FF6E24F6F33]

(No symbol) [0x00007FF6E24C0358]

(No symbol) [0x00007FF6E24C10C3]

GetHandleVerifier [0x00007FF6E297BA8D+3001453]

GetHandleVerifier [0x00007FF6E2975E72+2977874]

GetHandleVerifier [0x00007FF6E299497D+3103581]

GetHandleVerifier [0x00007FF6E26CC7EA+186826]

GetHandleVerifier [0x00007FF6E26D43FF+218591]

GetHandleVerifier [0x00007FF6E26B9D94+110452]

GetHandleVerifier [0x00007FF6E26B9F42+110882]

GetHandleVerifier [0x00007FF6E26A0379+5465]

BaseThreadInitThunk [0x00007FF807FCE8D7+23]

RtlUserThreadStart [0x00007FF80953BF6C+44]

**Copilot said:**

**Copilot**

El error "no such element: Unable to locate element" indica que Selenium no pudo encontrar el elemento especificado. Esto puede deberse a que el selector utilizado no es correcto o que el elemento no está presente en el DOM en el momento de la búsqueda.

Para solucionar este problema, vamos a verificar y ajustar los selectores utilizados para encontrar los íconos rojos de error y el mensaje de error. Aquí tienes una versión modificada del script:

**Paso 9: Ajustar Selectores y Validar Errores**

1. **Actualizar el script**: Modifica el script para utilizar selectores correctos y verificar la presencia de los íconos rojos de error y el mensaje de error.

**Paso 10: Ejecutar el Script**

1. **Ejecutar el script**: En Visual Studio Code, abre la terminal y navega hasta el directorio donde guardaste saucedemo\_login.py. Luego, ejecuta el script con el siguiente comando:

Este script ingresará un usuario y contraseña inválidos, verificará la presencia de los íconos rojos de error junto a los campos de usuario y contraseña, y comprobará que el mensaje de error debajo del campo de contraseña sea "Epic sadface: Username and password do not match any user in this service". Si todas las verificaciones son correctas, verás los mensajes correspondientes en la terminal.

¿Te gustaría probar este ejemplo y ver si funciona correctamente? ¡Avísame si necesitas más ayuda o si estás listo para continuar con otro escenario!

**1**Favicon typeWhat is the best way to avoid NoSuchElementException in Selenium?**2**Favicon typeorg.openqa.selenium.NoSuchElementException: no such elementFavicon typeNo Such Element Exception in Selenium: How Do You Handle It ...

Show more

Edit in Pages

**Copy**

AI-generated content may be incorrect

**You said:**

recuerda primero se ingresa el password, luego usuario incorrecto despues dar click en el boton de login.

hay un label abajo de password que menciona "Epic sadface: Username and password do not match any user in this service"

Olvidemonos de las cruces rojas que aparecen en alado de user y password. Solo que aparesca el mensaje abajo de password. debemos de permanecer en : https://www.saucedemo.com/

**Copilot said:**

**Copilot**

¡Entendido! Vamos a realizar la prueba ingresando primero la contraseña, luego el usuario incorrecto, y finalmente haciendo clic en el botón de login. Verificaremos que aparezca el mensaje de error debajo del campo de contraseña y que permanezcamos en la página de login.

**Paso 11: Automatizar el Escenario de Usuario y Contraseña Inválidos con Validación de Mensaje de Error**

1. **Actualizar el script**: Modifica el script para ingresar primero la contraseña, luego el usuario incorrecto, y verificar la presencia del mensaje de error.

**Paso 12: Ejecutar el Script**

1. **Ejecutar el script**: En Visual Studio Code, abre la terminal y navega hasta el directorio donde guardaste saucedemo\_login.py. Luego, ejecuta el script con el siguiente comando:

Este script ingresará primero la contraseña, luego el usuario incorrecto, verificará la presencia del mensaje de error debajo del campo de contraseña, y comprobará que permanezcamos en la página de login (https://www.saucedemo.com/). Si todas las verificaciones son correctas, verás los mensajes correspondientes en la terminal.

¿Te gustaría probar este ejemplo y ver si funciona correctamente? ¡Avísame si necesitas más ayuda o si estás listo para continuar con otro escenario!