

सेलेनियम के साथ वेब ब्राउज़र स्वचालन

सेलेनियम वेब ब्राउज़रों को स्वचालित करने का एक शक्तिशाली उपकरण है। यह आपको प्रोग्रामेटिक रूप से एक ब्राउज़र को नियंत्रित करने की अनुमति देता है ताकि वेब पेजों पर नेविगेट करने, फॉर्म भरने, बटन क्लिक करने और डेटा निकालने जैसे कार्य किए जा सकें। यह वेब स्क्रैप्टिंग, वेब एप्लिकेशन का परीक्षण करने और दोहराव वाले कार्यों को स्वचालित करने सहित विभिन्न कार्यों के लिए उपयोगी हो सकता है।

यहाँ पायथन के साथ सेलेनियम का उपयोग करके CSDN ब्लॉग को स्क्रैप्ट करने का एक बुनियादी उदाहरण दिया गया है:

```
from selenium import webdriver
from selenium.webdriver.chrome.options import Options
from selenium.webdriver.common.by import By
from selenium.common.exceptions import NoSuchElementException
import time

def scrape_csdn_blog(url):
    """
    CSDN
    (a)
    ,
    "https://blog.csdn.net/lzw_java/article"

Args:
    url (str): CSDN URL
    """

try:
    #
    chrome_options = Options()
    chrome_options.add_argument("--headless") #
    chrome_options.add_argument("--disable-gpu") # GPU
    chrome_options.add_argument("--no-sandbox") # OS
    chrome_options.add_argument("--disable-dev-shm-usage") #

    #
    driver = webdriver.Chrome(options=chrome_options)

    #
    driver.get(url)

    # 'a'
    links = driver.find_elements(By.TAG_NAME, 'a')

    if not links:
        print("")


```

```

        driver.quit()

    return

for link in links:
    try:
        href = link.get_attribute('href')
        if href and href.startswith("https://blog.csdn.net/lzw_java/article"):
            text = link.text.strip()

            print(f" : {text}")
            print(f"URL: {href}")
            print("-" * 20)

    except Exception as e:
        print(f" : {e}")
        continue

except Exception as e:
    print(f" : {e}")

finally:
    #
    if 'driver' in locals():
        driver.quit()

if __name__ == "__main__":
    blog_url = "https://blog.csdn.net/lzw_java?type=blog" # URL
    scrape_csdn_blog(blog_url)

```