

爬取CSDN博客

爬取CSDN博客是一种自动化操作，使用Python的Selenium库可以实现。Selenium是一个用于自动化浏览器操作的库，可以用于模拟用户行为，如登录、浏览、点击等。在本文中，我们将使用Selenium来爬取CSDN博客的内容。

爬取CSDN博客的步骤如下：

```
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):
    """
    爬取CSDN博客
    """
    # 初始化Chrome浏览器
    chrome_options = Options()
    chrome_options.add_argument("--headless") # 无头模式
    chrome_options.add_argument("--disable-gpu") # 禁用GPU
    chrome_options.add_argument("--no-sandbox") # 禁用sandbox
    chrome_options.add_argument("--disable-dev-shm-usage") # 禁用dev-shm-usage

    driver = webdriver.Chrome(options=chrome_options)
    driver.get(url)
    time.sleep(5)

    # 获取博客内容
    title = driver.find_element(By.CSS_SELECTOR, "h1").text
    content = driver.find_element(By.CSS_SELECTOR, "div").text

    # 打印博客内容
    print(title)
    print(content)

    # 关闭浏览器
    driver.quit()

# 爬取CSDN博客
url = "https://blog.csdn.net/lzw_java/article"
scrape_csdn_blog(url)
```

```

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)

```