MÃ NGUỒN MỞ TRONG KHOA HỌC DỮ LIỆU

Bài 02.
THU THẬP DỮ LIỆU SỬ DỤNG SELENIUM

ThS. Lê Nhật Tùng

1. Giới thiệu về Selenium

- Selenium là một framework mã nguồn mở tự động hóa trình duyệt web cho phép người dùng điều khiển các trình duyệt một cách tự động. Selenium thường được sử dụng để:
  - Thực hiện kiểm thử tự động cho các ứng dụng web.
  - Thu thập dữ liệu từ các trang web (Web Scraping).
  - Tự động hóa các tác vụ lặp đi lặp lại trên trình duyệt.

2. Cài đặt Selenium

- Đảm bảo rằng Python đã được cài đặt trên máy tính. Bạn có thể tải Python tại python.org.
- Mở terminal hoặc command prompt và gõ lệnh sau để cài đặt Selenium:

```
pip install selenium
```

2. Cài đặt Selenium

- Selenium yêu cầu một WebDriver cho trình duyệt mà bạn muốn tự động hóa.
- Dưới đây là một số WebDriver phổ biến:
  - Chrome: ChromeDriver
  - Firefox: GeckoDriver
  - Edge: Edge WebDriver
- Sau khi tải WebDriver, hãy thêm đường dẫn đến thư mục chứa WebDriver vào biến môi trường PATH.

3. Cấu trúc cơ bản của một script Selenium

3.1. Khởi tạo WebDriver

```
from selenium import webdriver

# Khởi tạo WebDriver
driver = webdriver.Chrome()  # Hoặc webdriver.Firefox() cho Firefox
```

3.2. Mở một trang web

```
driver.get("https://www.example.com")
```

3.3. Tương tác với trang web

Selenium cho phép bạn tương tác với các phần tử trên trang web.

Ví dụ, bạn có thể tìm và nhấp vào một nút:

```
# Tìm nút theo ID và nhấp vào
button = driver.find_element_by_id("my-button-id")
button.click()
```

3. Cấu trúc cơ bản của một script Selenium

3.4. Thu thập dữ liệu
Sau khi truy cập trang, bạn có thể thu thập dữ liệu bằng cách tìm kiếm các phần tử và lấy thông tin từ chúng:

```
# Lấy tiêu đề của trang
title = driver.title
print("Tiêu đề trang:", title)

# Lấy nội dung của một phần tử
element = driver.find_element_by_class_name("my-class")
content = element.text
print("Nội dung:", content)
```

4. Giới thiệu về XPath

- XPath (XML Path Language) là một ngôn ngữ dùng để định vị các phần tử trong tài liệu XML và HTML. Trong Selenium, XPath rất hữu ích để xác định các phần tử mà bạn muốn tương tác. Bạn có thể sử dụng XPath để truy cập bất kỳ phần tử nào trên trang web, ngay cả khi nó không có ID hoặc class.

4. Giới thiệu về XPath

XPath cho phép bạn sử dụng cú pháp để xác định các phần tử theo nhiều cách khác nhau:

Chọn tất cả các phần tử:

```
//div
```

Câu lệnh trên sẽ chọn tất cả các phần tử trên trang.

4. Giới thiệu về XPath

XPath cho phép bạn sử dụng cú pháp để xác định các phần tử theo nhiều cách khác nhau:

```
//input[@type='text']
```

Câu lệnh trên sẽ chọn tất cả các phần tử <input> có thuộc tính type bằng text.

4. Giới thiệu về XPath

XPath cho phép bạn sử dụng cú pháp để xác định các phần tử theo nhiều cách khác nhau:

Chọn phần tử con:

//div/p

Câu lệnh trên sẽ chọn tất cả các phần tử <p> bên trong một phần tử .

5. Các phương thức quan trọng

- Các phương thức định vị phần tử (find_element)
- Trước đây, chúng ta thường sử dụng các phương thức như find_element_by_id, find_element_by_xpath,... Tuy nhiên, từ phiên bản Selenium 3 trở đi, cách viết này đã được đơn giản hóa và rõ ràng hơn.

```
from selenium.webdriver.common.by import By
element = driver.find_element(By.ID, "my_element_id")
```

5. Các phương thức quan trọng

- **Các kiểu định vị phổ biến khác:**
  - **By.NAME:** Định vị theo thuộc tính name
  - **By.XPATH:** Định vị theo XPath
  - **By.CLASS_NAME:** Định vị theo class name
  - **By.TAG_NAME:** Định vị theo tag name
  - **By.LINK_TEXT:** Định vị theo văn bản của link
  - **By.PARTIAL_LINK_TEXT:** Định vị theo một phần văn bản của link
  - **By.CSS_SELECTOR:** Định vị theo CSS selector

5. Các phương thức quan trọng

Các phương thức tương tác với phần tử

- **click():** Nhấp vào phần tử
- **send_keys("text"):** Nhập văn bản vào phần tử
- **clear():** Xóa nội dung của phần tử
- **submit():** Gửi form
- **get_attribute("attribute"):** Lấy giá trị của một thuộc tính
- **is_displayed():** Kiểm tra xem phần tử có hiển thị hay không
- **is_enabled():** Kiểm tra xem phần tử có thể tương tác được hay không
- **is_selected():** Kiểm tra xem phần tử (checkbox, radio button) có được chọn hay không

5. Các phương thức quan trọng

Các phương thức điều khiển trình duyệt

- **get("url"):** Mở một URL
- **back():** Quay lại trang trước
- **forward():** Đi tới trang tiếp theo
- **refresh():** Làm mới trang
- **title():** Lấy tiêu đề của trang
- **current_url():** Lấy URL hiện tại
- **page_source():** Lấy toàn bộ HTML của trang
- **execute_script("javascript"):** Thực thi một đoạn JavaScript

5. Các phương thức quan trọng

Các phương thức khác

- **switch_to.frame():** Chuyển đổi sang một iframe
- **switch_to.window():** Chuyển đổi sang một cửa sổ khác
- **switch_to.alert():** Chuyển đổi sang một hộp thoại alert
- **implicitly_wait(time):** Thiết lập thời gian chờ ngầm định khi tìm kiếm phần tử
- **WebDriverWait():** Đợi cho một điều kiện nào đó xảy ra trước khi tiếp tục

Thực hành 1

```python
from pygments.formatters.html import webify
from selenium import webdriver
from selenium.webdriver.common.by import By
import time

# Khởi tạo Webdriver
driver = webdriver.Chrome()

# Mở trang
url = "https://en.wikipedia.org/wiki/List_of_painters_by_name"
driver.get(url)

# Đợi khoảng chừng 2 giây
time.sleep(2)

# Lay tat ca cac the <a>
tags = driver.find_elements(By.TAG_NAME, "a");

# Tao ra danh sach cac lien ket
links = [tag.get_attribute("href") for tag in tags]

# Xuat thong tin
for link in links:
    print(link)

# Dong webdriver
driver.quit()
```

Thực hành 2

```python
from selenium import webdriver
from selenium.webdriver.common.by import By
import time

# Khởi tạo Webdriver
driver = webdriver.Chrome()

# Mở trang
url = "https://en.wikipedia.org/wiki/List_of_painters_by_name"
driver.get(url)

# Đợi khoảng chừng 2 giây
time.sleep(2)

# Lay tat ca cac the <a> voi title chua "List of painters"
tags = driver.find_elements(By.XPATH, "//a[contains(@title, 'List of painters')]")

# Tao ra danh sach cac lien ket
links = [tag.get_attribute("href") for tag in tags]

# Xuat thong tin
for link in links:
    print(link)

# Dong webdriver
driver.quit()
```

Thực hành 3

```
from selenium import webdriver
from selenium.webdriver.common.by import By
import time

# Khởi tạo Webdriver
driver = webdriver.Chrome()

# Mở trang
url = "https://en.wikipedia.org/wiki/List of painters by name beginning with %22P%22"
driver.get(url)

# Đợi một chút để trang tải
time.sleep(2)

# Lay ra tat cac ca the ul
ul_tags = driver.find_elements(By.TAG_NAME, "ul")
print(len(ul_tags))

# Chon the ul thu 21
ul_painters = ul_tags[20] # list start with index=0

# Lay ra tat ca the <li> thuoc ul_painters
li_tags = ul_painters.find_elements(By.TAG_NAME, "li")
```

Thực hành 3 (tt)

```
# Chon the ul thu 21
ul_painters = ul_tags[20] # list start with index=0

# Lay ra tat ca the <li> thuoc ul_painters
li_tags = ul_painters.find_elements(By.TAG_NAME, "li")

# Tao danh sach cac url
links = [tag.find_element(By.TAG_NAME, "a").get_attribute("href") for tag in li_tags]

# Tao danh sach cac url
titles = [tag.find_element(By.TAG_NAME, "a").get_attribute("title") for tag in li_tags]

# In ra url
for link in links:
    print(link)

# In ra title
for title in titles:
    print(title)

# Dong webdrive
driver.quit()
```

Thực hành 4

```python
from builtins import range
from selenium import webdriver
from selenium.webdriver.common.by import By
import time

# Khởi tạo Webdriver
driver = webdriver.Chrome()

for i in range(65, 91):
    url = "https://en.wikipedia.org/wiki/List_of_painters_by_name_beginning_with_%22"+chr(i)+"%22"
    try:
        # Mở trang

        driver.get(url)

        # Đợi một chút để trang tải
        time.sleep(3)

        # Lay ra tat cac ca the ul
        ul_tags = driver.find_elements(By.TAG_NAME, "ul")
        print(len(ul_tags))

        # Chon the ul thu 21
        ul_painters = ul_tags[20]  # list start with index=0

        # Lay ra tat ca the <li> thuoc ul_painters
        li_tags = ul_painters.find_elements(By.TAG_NAME, "li")

        # Tao danh sach cac url
        titles = [tag.find_element(By.TAG_NAME, "a").get_attribute("title") for tag in li_tags]

        # In ra title
        for title in titles:
            print(title)
    except:
        print("Error!")

# Dong webdrive
driver.quit()
```

Thực hành 5

```
from pygments.formatters.html import webify
from selenium import webdriver
from selenium.webdriver.common.by import By
import time
import pandas as pd
import re

# Tao dataframe rong
d = pd.DataFrame({'name': [], 'birth': [], 'death': [], 'nationality':[]})

# Khoi tao webdriver
driver = webdriver.Chrome()

# Mo trang
url = "https://en.wikipedia.org/wiki/Edvard_Munch"
driver.get(url)

# Doi 2 giay
time.sleep(2)
```

Thực hành 5

```
# Lay ten hoa si
try:
    name = driver.find_element(By.TAG_NAME, "h1").text
except:
    name = ""
# Lay ngay sinh
try:
    birth_element = driver.find_element(By.XPATH, "//th[text()='Born']/following-sibling::td")
    birth = birth_element.text
    birth = re.findall(r'[0-9]{1,2}+\s+[A-Za-z]+\s+[0-9]{4}', birth)[0] # regex
except:
    birth = ""
# Lay ngay mat
try:
    death_element = driver.find_element(By.XPATH, "//th[text()='Died']/following-sibling::td")
    death = death_element.text
    death = re.findall(r'[0-9]{1,2}+\s+[A-Za-z]+\s+[0-9]{4}', death)[0]
except:
    death = ""
# Lay ngay mat
try:
    nationality_element = driver.find_element(By.XPATH, "//th[text()='Nationality']/following-sibling::td")
    nationality = nationality_element.text
except:
    nationality = ""
```

Thực hành 5

```
# Tao dictionary thong tin cua hoa si
painter = {'name' : name, 'birth': birth, 'death': death, 'nationality':nationality}

# CHuyen doi dictionary thanh DataFrame
painter_df = pd.DataFrame([painter])

# Them thong tin vao DF chinh
d = pd.concat([d, painter_df], ignore_index=True)

# In ra DF
print(d)

# Dong web driver
driver.quit()
```

Thực hành 6

```
from pygments.formatters.html import webify
from selenium import webdriver
from selenium.webdriver.common.by import By
import time
import pandas as pd
import re

##################################################
# I. Tai noi chua links vaf Tao dataframe rong
all_links = []
d = pd.DataFrame({'name': [], 'birth': [], 'death': [], 'nationality':[]})
```

Thực hành 6

```
#########################################################
# II. Lay ra tat ca duong dan de truy cap den painters
# Khởi tạo Webdriver
for i in range(70, 71):
    driver = webdriver.Chrome()
    url = "https://en.wikipedia.org/wiki/List_of_painters_by_name_beginning_with_%22"+chr(i)+"%22"
    try:

        # Mở trang
        driver.get(url)

        # Đợi một chút để trang tải
        time.sleep(3)

        # Lay ra tat cac ca the ul
        ul_tags = driver.find_elements(By.TAG_NAME, "ul")
        print(len(ul_tags))

        # Chon the ul thu 21
        ul_painters = ul_tags[20]  # list start with index=0

        # Lay ra tat ca the <li> thuoc ul_painters
        li_tags = ul_painters.find_elements(By.TAG_NAME, "li")

        # Tao danh sach cac url
        links = [tag.find_element(By.TAG_NAME, "a").get_attribute("href") for tag in li_tags]
        for x in links:
            all_links.append(x)
    except:
        print("Error!")

    # Dong webdrive
    driver.quit()
```

Thực hành 6

```
#########################################################
# III. Lay thong tin cua tung hoa si
count =0;
for link in all_links:
    if (count>3):
        break
    count=count+1;

    print(link)
    try:
        # Khoi tao webdriver
        driver = webdriver.Chrome()
        # Mo trang
        url = link
        driver.get(url)

        # Doi 2 giay
        time.sleep(2)

        # Lay ten hoa si
        try:
            name = driver.find_element(By.TAG_NAME, "h1").text
        except:
            name = ""
        # Lay ngay sinh
        try:
            birth_element = driver.find_element(By.XPATH, "//th[text()='Born']/following-sibling::td")
            birth = birth_element.text
            birth = re.findall(r'[0-9]{1,2}+\s+[A-Za-z]+\s+[0-9]{4}', birth)[0] # regex
        except:
            birth = ""
```

Thực hành 6

```
        # Lay ngay mat
        try:
            nationality_element = driver.find_element(By.XPATH, "//th[text()='Nationality']/following-sibling::td")
            nationality = nationality_element.text
        except:
            nationality = ""

        # Tao dictionary thong tin cua hoa si
        painter = {'name' : name, 'birth': birth, 'death': death, 'nationality':nationality}

        # CHuyen doi dictionary thanh DataFrame
        painter_df = pd.DataFrame([painter])

        # Them thong tin vao DF chinh
        d = pd.concat([d, painter_df], ignore_index=True)

        # Dong web driver
        driver.quit()
    except:
        pass

####################
# IV. In thong tin
print(d)
# determining the name of the file
file_name = 'Painters.xlsx'

# saving the excel
d.to_excel(file_name)
print('DataFrame is written to Excel File successfully.')
```