باسمه تعالی

درس: بازیابی اطلاعات

نام و نام خانوادگی: تینا توکلی و هادی امینی

شماره دانشجویی: 9922762220و 9912762270

شماره تمرین :03

توضیح کد :

از کتابخانه selenium برای استخراج داده از وب سایت IEEE Xplore استفاده می کنیم.

متغیر سراسری:

driver: یک متغیر سراسری برای نگهداری از نمونه WebDriver. در ابتدا به None تنظیم می شود.

توابع:

load_homepage:()

یک نمونه جدید Chrome WebDriver ایجاد می کند.

به URL صفحه اصلی IEEE Xplore هدایت می شود.

نمونه WebDriver را برمی گرداند.

search(query):

منتظر می ماند تا فیلد ورودی جستجو قابل کلیک باشد.

کادر جستجو را پیدا می کند و query ارائه شده را وارد می کند.

دکمه جستجو را پیدا می کند و روی آن کلیک می کند.

get_papers:()

یک لیست خالی paper_list برای ذخیره لینک های مقاله ایجاد می کند.

5 بار حلقه می زند (قابل تنظیم):

منتظر حضور نتایج جستجو می ماند.

تمام عناصر با کلاس result-item-align را پیدا می کند.

هر مورد نتیجه را پیمایش می کند:

publisher-info-container را پیدا می کند.

نوع مقاله را با استفاده از نام های تگ و متن از دومین عنصر span استخراج می کند.

اگر نوع "مقاله کنفرانس" باشد:

لینک مقاله را با استفاده از ویژگی href آن استخراج می کند.

لینک را به paper_list اضافه می کند.

next_page() را برای کلیک بر روی دکمه "صفحه بعد" (در صورت وجود) فراخوانی می کند.

paper_list را برمی گرداند.

next_page:()

ظرف دکمه "صفحه بعد" (next-page-set) را پیدا می کند.

منتظر می ماند تا دکمه قابل کلیک باشد.

از اجرای جاوا اسکریپت (execute_script) برای کلیک بر روی دکمه استفاده می کند (مشکلات احتمالی تعامل را مدیریت می کند).

select_value(string):

رشته ورودی را با استفاده از جداکننده ": " تقسیم می کند و مقدار بعد از دو نقطه را برمی گرداند (فرمت خاصی برای استخراج داده فرض می کند).

extract_data(paper_link):

به paper_link ارائه شده هدایت می شود.

منتظر می ماند تا عنصر عنوان مقاله موجود باشد.

یک دیکشنری خالی info برای ذخیره داده های استخراج شده برای این مقاله ایجاد می کند.

اطلاعات مختلف را با استفاده از عبارات XPath، نام کلاس ها و انتخاب کننده های CSS خاص استخراج می کند (برخی در بلوک های try...except برای مدیریت خطاهای احتمالی پیچیده شده اند):

سلکتورها:

1. title:

- **انتخاب کننده:** 'By.CLASS_NAME, 'document-title
- **توضیح:** این سلکتور به دنبال عنصری با نام کلاس document-title می‌گردد. این نام کلاس احتمالاً به عنصری اشاره می‌کند که عنوان سند را در صفحه وب نشان می‌دهد.

2. pages:

- **انتخاب کننده:** "By.XPATH, "//div[@class='u-pb-1'][strong[contains(text(), 'Page(s):')]]
- **توضیح:** این عبارت XPath یک ساختار عنصری خاص را هدف قرار می‌دهد:
  - //div[@class='u-pb-1']: عناصر div را با نام کلاس u-pb-1 پیدا می‌کند. این کلاس ممکن است به بخشی حاوی اطلاعات صفحه مربوط باشد.

- [strong[contains(text(), 'Page(s):')]]: عناصر div را فقط به مواردی که شامل یک عنصر strong با متنی حاوی "Page(s):" هستند، فیلتر می‌کند. این نشان می‌دهد که عنصر strong برچسب شماره صفحه را نگه می‌دارد.

3.Cites in Papers :

- **انتخاب کننده:** By.XPATH, "//button[div[text()='Papers']]//div[@class='document-banner-metric-count']"
- **توضیح:** این عبارت XPath یک عنصر دکمه را هدف قرار می‌دهد که به Cites in Papers مربوط می‌شود:
  - //button[div[text()='Papers']]: دکمه‌هایی را پیدا می‌کند که شامل یک عنصر div با متن "Papers" هستند. این نشان می‌دهد که دکمه به Cites in Papers مربوط می‌شود.
  - //div[@class='document-banner-metric-count']: در داخل آن دکمه‌ها، به دنبال یک عنصر div با کلاس document-banner-metric-count می‌گردد. این کلاس احتمالاً تعداد واقعی استنادات را نگه می‌دارد.

4.Cites in Patent :

- **انتخاب کننده:** By.XPATH, "//div[contains(@class, 'document-banner-metric-container')]//button[div[text()='Cites in'][following-sibling::div[text()='Patent']]]/div[@class='document-banner-metric-count']"
- **توضیح:** این عبارت XPath پیچیده یک عنصر دکمه را هدف قرار می‌دهد که به Cites in Patent مربوط می‌شود:
  - //div[contains(@class, 'document-banner-metric-container')]: یک عنصر div را پیدا می‌کند که نام کلاس آن شامل "document-banner-metric-container" است. این کلاس ممکن است به یک ظرف برای معیارهای سند مربوط باشد.
  - //button[div[text()='Cites in'][following-sibling::div[text()='Patent']]]: در داخل آن ظرف، به دنبال دکمه‌هایی با یک div حاوی "Cites in" می‌گردد که به دنبال آن یک div هم‌تراز با "Patent" وجود دارد. این ساختار نشان می‌دهد که دکمه به طور خاص برای Cites in Patent است.
  - /div[@class='document-banner-metric-count']: در نهایت، عنصر div را در داخل دکمه انتخاب می‌کند که احتمالاً تعداد Cites را نگه می‌دارد.

5. Full Text Views :

- **انتخاب کننده:** By.XPATH, "//button[div[div[text()='Full']]]/div[@class='document-banner-metric-count']"
- **توضیح:** این عبارت XPath یک عنصر دکمه را پیدا می‌کند که به بازدیدهای متن کامل مربوط می‌شود:
  - //button[div[div[text()='Full']]]: دکمه‌هایی را پیدا می‌کند که شامل یک ساختار div تو در تو با div دیگری حاوی متن "Full" هستند. این نشان می‌دهد که دکمه به دسترسی به متن کامل مربوط می‌شود.
  - /div[@class='document-banner-metric-count']: عنصر div را در داخل دکمه انتخاب می‌کند که احتمالاً تعداد بازدیدهای متن کامل را نگه می‌دارد.

6. Publisher :

- انتخاب کننده : By.XPATH, '//span[contains(By.CLASS_NAME,"publisher-info-container")

- توضیح (تلاش 2): این عبارت XPath جایگزین دیگری را ارائه می‌دهد که یک ساختار span تو در تو را در داخل عنصری با کلاس حاوی "publisher-info-container" هدف قرار می‌دهد.

7. DOI :

- انتخاب کننده: 'By.CLASS_NAME, 'stats-document-abstract-doi
- توضیح: این سلکتور به دنبال عنصری با نام کلاس stats-document-abstract-doi می‌گردد. این نام کلاس احتمالاً به عنصری اشاره می‌کند که شناسه شیء دیجیتال (DOI) را در خود جای داده است.

8. Date of Publication :

- انتخاب کننده: 'By.CLASS_NAME, 'doc-abstract-confdate
- توضیح: این سلکتور به دنبال عنصری با نام کلاس doc-abstract-confdate می‌گردد. این نام کلاس احتمالاً به عنصری اشاره می‌کند که Date of Publication را در خود جای داده است.

9. Abstract :

- انتخاب کننده: '[g-0"]//div[@xplmathjax row By.XPATH, '//div[@class="abstract-text
- توضیح: این عبارت XPath یک عنصر div را با کلاس g-0 row abstract-text و حاوی یک عنصر div با ویژگی xplmathjax هدف قرار می‌دهد. این ساختار احتمالاً به عنصری اشاره می‌کند که Abstract را در خود جای داده است.

10. Published in :

- انتخاب کننده: 'a By.CSS_SELECTOR, '.stats-document-abstract-publishedIn
- توضیح: این سلکتور CSS به دنبال عنصر a با کلاس stats-document-abstract-publishedIn می‌گردد. این عنصر احتمالاً پیوند به نشریه‌ای را که سند در آن منتشر شده است، در خود جای داده است.

11. Authors :

- **توضیح:**
  - این کد ابتدا عنصر دکمه با شناسه authors را پیدا می‌کند و بر روی آن کلیک می‌کند.
  - سپس تمام عناصر div با کلاس author-card را پیدا می‌کند.
  - برای هر عنصر author-card، متن آن را به یک دیکشنری با کلیدهای name و from (در صورت وجود) تبدیل می‌کند.
  - در نهایت، لیست دیکشنری‌های نویسنده را در info ذخیره می‌کند.

12. IEEE Keywords:

- **توضیح:**
  - این کد ابتدا عنصر دکمه با شناسه keywords را پیدا می‌کند و بر روی آن کلیک می‌کند.
  - سپس تمام عناصر a را با داده tealium_data حاوی "IEEE Keywords" پیدا می‌کند.
  - در نهایت، لیست متن عناصر a را در info با عنوان Keywords IEEE ذخیره می‌کند.

13. Author Keywords:

- **توضیح:**
  - این کد ابتدا عنصر دکمه با شناسه keywords را پیدا می‌کند و بر روی آن کلیک می‌کند.
  - سپس تمام عناصر a را با داده tealium_data حاوی "Author Keywords" پیدا می‌کند.
  - در نهایت، لیست متن عناصر a را در info با عنوان Keywords Author ذخیره می‌کند.

و در انتها دیکشنری info حاوی داده های استخراج شده را برمی گرداند وآنها را با save_json() ذخیره میکند .

چالش ها:

1. در بخش Author Keywords : لیست خالی Author Keywords

با استفاده از XPath به دنبال عناصری با `data-tealium_data` حاوی "Author Keywords" هستیم و سپس متن آنها را در لیست `Author Keywords` ذخیره می‌کنیم. ولی در برخی موارد ممکن است tagهای اضافی را انتخاب میکرد که دارای هیچ متنی نیستند .

راه حل:

با استفاده از شرط `len(e.text) > 0` این موضوع را هندل کردیم که به معنی فقط در صورتی که عنصر `e` دارای متن باشد، آن را به لیست اضافه می‌کنیم.

2. **کلیک Newest:**

تأخیر قابل توجهی در کلیک روی دکمه "Newest" برای مرتب سازی نتایج وجود دارد.

علت:

دلایل احتمالی متعددی برای این تأخیر وجود دارد:

1. **بارگذاری ناهمزمان:** ممکن است عناصر صفحه وب به طور ناهمزمان بارگذاری شوند، به این معنی که دکمه "Newest" ممکن است بلافاصله پس از بارگذاری صفحه قابل کلیک نباشد.
2. **پیچیدگی صفحه:** اگر صفحه وب از نظر ساختاری پیچیده باشد، ممکن است یافتن دکمه "Newest" برای Selenium زمان بیشتری طول بکشد.
3. **تأخیر شبکه:** اگر اتصال شبکه ضعیف باشد، ممکن است ارسال درخواست کلیک به دکمه "Newest" به سرور زمان بیشتری طول بکشد.

راه حل:

1. **استفاده از WebDriverWait:** از کتابخانه `WebDriverWait` برای صبر کردن تا دکمه "Newest" قابل کلیک شود قبل از کلیک بر روی آن استفاده شده است. این کار با استفاده از روش `until` انجام می شود که

یک شرط را به عنوان ورودی می پذیرد. در این مورد، شرط این است که دکمه با استفاده از `EC.element_to_be_clickable` قابل کلیک باشد.

2. **استفاده از `try...except`:** در نهایت، از یک بلوک `try...except` برای تلاش برای کلیک بر روی دکمه "Newest" با استفاده از روش `click` استفاده شده است. اگر این کار به دلیل `ElementNotInteractableException` شکست بخورد، از `driver.execute_script` برای کلیک بر روی دکمه با استفاده از JavaScript استفاده می شود. این روش می تواند در مواردی مفید باشد که Selenium قادر به کلیک بر روی دکمه با استفاده از روش های معمول نیست.

3. **وجود نداشتن برخی موارد در برخی مقالات:**

برخی از صفحات وب ممکن است برخی عناصر را نداشته باشند. این امر می تواند منجر به خطا در هنگام تلاش برای یافتن این عنصر شود.

راه حل:

برای حل این مشکل، از یک بلوک `try...except` استفاده شده است. در داخل بلوک `try`، کد تلاش می کند عناصر را با استفاده ازselector ارائه شده پیدا کند و مقدارآنرا تنظیم کند. اگر این کار به دلیل خطا (مانند `ElementNotFoundException`) شکست بخورد، `None` تنظیم می شود.

خروجی:

شامل دو فایل با پسوند json است که یکی برای relevance و newest است که داخل هر کدام شامل تعدادی مقاله کنفرانسی هست.

بخش اضافی:

یک رابط کاربری تحت وب برای نمایش محتویات فایلهای خروجی ایجاد کردیم.

Sort By

Relevance

Title	Assessment of Information Security in Integrated Systems
DOI	10.1109/ITQMIS53292.2021.9642824
Date of Publication	06-10 September 2021
Full Text Views	1302
Publisher	IEEE
Published in	2021 International Conference on Quality Management, Transport and Information Security, Information Technologies (IT&QM&IS)
Page(s)	None
Cites in Papers	None
Cites in Patent	None
Authors	Tatyana Y. Khashirova (Institute of Informatics, Electronics and Robotics, Kabardino-Balkarian State University named after H.M. Berbekov, Nalchik, Russia) Ibragim I. Mamuchiev (Institute of Informatics, Electronics and Robotics, Kabardino-Balkarian State University named after H.M. Berbekov, Nalchik, Russia) Madina I. Mamuchieva (Institute of Informatics, Electronics and Robotics, Kabardino-Balkarian State University named after H.M. Berbekov, Nalchik, Russia)
Abstract	The article discusses the typesetting and structural model of an information protection system. An algorithm for analyzing information security based on a structural model of an information security system is also presented. The goal of the work is to develop an information security
IEEE Keywords	Analytical models, Typesetting, Computational modeling, Information security, Linear programming, Computer security, Information systems
Author Keywords	information security, assessment of the security of the information security system, integrated system, computer modelling, information security systems

Previous Next

Paper 4 of 110

مشارکت:

امینی	10
توکلی	10