2023. 주제탐구 최종보고서

Python selenium을 이용한 웹 크롤링 자동화 프로그램 개발

연 구 자: 이 재 준(김해분성고등학교 2학년)
이 상 민(김해분성고등학교 2학년)
남 형 준(김해분성고등학교 1학년)
정 우 혁(김해분성고등학교 1학년)

지도교사: 안 지 연(김해분성고등학교)

< 목 차 >

1. 서 론

가. 연구 목적

○ 웹사이트 내에서 반복해왔던 단순한 과정을 자동으로 수행하는 프로그램을 개발하여 반복적인 업무를 효율적으로 수행할 수 있게 하고 인터넷을 이용할 때 겪을 수 있는 사소한 번거로움을 해결하고자 한다.

나. 연구 주제의 선정

○ 때때로 웹사이트를 접속할 때면 로그인이나 인증을 해야 하는 경우가 있다. 반복된 메일을 보내야 하는 경우나 온라인으로 출석 체크를 하는 경우, 특정 시간에 수강 신청하는 경우, 공지를 확인해야 할 때 같은 주소의 웹사이트에 반복하여 접속하는 경우가 종종 있다. 앞의 상황은 모두 반복적이거나 단순한 작업이고 이것들을 하는 데에 시간을 보내는 것이 다소 지루하게 느껴지거나 귀찮다는 생각이 들기도 한다. 따라서 이러한 과정을 자동화하여 인터넷을 조금 더 편리하게 사용할 방법을 탐색하게 되었다. 많은 문제 중에서 교내 공지사항 접근 문제로 불편 사항을 구체화했고 웹 크롤링으로 주제를 정하였다.

2. 본 론

가. 이론적 배경 및 선행 연구

○ 연구에 사용되는 언어는 Python이다. Python은 쉽고 빠른 개발이 가능하고 라이브러리 활용성, 확장성 및 재사용을 극대화하고자 만든 언어이다. 유니코드 문자열을 지원해서 다양한 국가 언어의 문자 처리에 매우 유용하게 활용되고 있으며 과학, 공학 분야에서도 빈번히 활용되고 있다.

○ 본 연구에서 사용하는 Python 프레임워크는 Selenium이다. 프레임워크는 개발자가 소프트웨어 개발을 가속하여 프로덕션 배포에 이르는 데 도움이 되는 유연한 범위의 소프트웨어 구성 요소를 제공한다. Selenium은 위에서 언급한 Python 프레임워크의 한 종류로 브라우저를 자동화하는 데에 사용된다. Selenium을 사용하면 웹 크롤링이나 메일 자동 송신 등 브라우저 내에서 자동으로 업무를 처리할 수 있다. 현재 BeautifulSoup, Scrapy 그리고 Selenium 등 웹 크롤링을 수행하는 여러 라이브러리가 존재하지만, 이 중에서 Selenium을 선택한 이유는 Selenium은 클릭과 같은 상호작용을 자동화하는 능력에 강점이 있기 때문이다.

○ 웹 크롤링은 방대한 웹 문서를 제공하는 웹에서 특정 웹사이트의 웹 문서를 자동으로 수집하는 일을 말한다. 자동으로 수집한 웹 문서를 이메일로 보내기 위해서는 smtplib를 사용한다. SMTP란 인터넷에서 이메일을 보내기 위해 이용되는 프로토콜이며 파이썬에서는 smtplib를 사용해 이메일을 송, 수신한다.

나. 연구 방법

○ 문제 해결을 위해 웹 크롤링 Python 프레임워크인 Selenium을 이용하는 방법을 구상하였다. 연구를 연구1과 연구2로 나누어 진행하였고 기본적인 연구는 Visual Studio Code, Python, 프레임워크가 설치 및 import 되어있는 Windows10 환경에서 진행되었다. 본 연구에서 수집한 정보는 김해분성고등학교 홈페이지 (https://bunsung-h.gne.go.kr/bunsung-h/main.do)의 공지이며 연구 기간은 2023년 8월 1일부터 11월 15일까지이다.

다. 연구 활동 및 과정

○ 연구가 진행된 환경은 다음과 같다.

운영체제: Windows10
IDE: Visual Studio Code 버전 1.8x,
사용 언어: Python 버전 3.11.4,
프레임워크: Selenium 버전 4.11.2
브라우저: Chrome 버전 115.0.5790.110 (64비트)

먼저 본 연구를 연구1과 연구2로 나누었다. 연구1에서는 Selenium 프레임워크를 이용해 웹 페이지를 자동으로 탐색하여 웹 문서를 수집하는 기능을 구현할 것이고 연구2에서는 연구1에서 수집한 웹 문서를 이메일로 자동 전송하는 기능을 구현할 것이다. 본 연구에서 수집한 정보는 본교 홈페이지 최상단 공지의 내용과 후속 공지의 제목이다.

Fig. 1. 최상단 공지 제목에 해당하는 XPath 복사

○ 연구1

Selenium을 이용해 만든 프로그램이 웹 페이지를 탐색해 정보를 수집하는 기능을 할 것이라는 가설을 세운 뒤 연구를 진행했다. 우선 위에서 언급한 환경에 맞게 연구 환경을 설정해 주고 “.py”파일을 만든다. 파일명은 “webC”로 하였다. 그다음 Selenium을 import 해준다. 이어서 Chrome driver를 로드해주고 목표로 하는 기능을 구현하기 위해 코드를 작성해 나간다. 이때 웹 페이지가 로드되는 시간을 고려해서 최대 20~30초까지 기다리도록 설정하였다. 정보를 수집할 개체(공지 내용, 제목)의 class 또는 id 명을 직접 설정하려 했으나 코드가 너무 복잡해졌고 제대로 작동되지 않았기 때문에 XPath를 이용했다. 공지 제목과 공지 내용, 후속 공지 내용 위치에 해당하는 XPath를 복사하여 각각을 elemHead, elemBody, elemBody_1에 저장하고 그것들의 텍스트

요소를 가져온다. 또 후속 공지를 클릭하는 동작을 추가하여 후속 공지로 이동한 뒤에 현재 페이지의 URL을 가져왔다. 가져온 URL을 current_url 변수에 추가한 뒤 Info에 f-string을 사용해 모든 변수를 입력하였고 각각의 내용들을 쉽게 구분하기 위해 줄 바꿈과 설명을 추가해주었다. 마지막으로 Chrome 브라우저를 닫는 동작을 추가해주어 마무리하였다.

○ 연구2

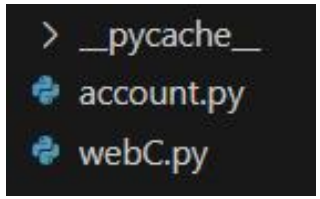

Fig. 2.
Visual Studio Code 내에서 webC.py와 account.py의 모습

수집한 공지 사항의 정보를 미리 입력해 놓은 이메일 주소로 자동 전송할 것이라는 가설을 세운 뒤 연구를 진행했다. 본격적인 연구에 앞서 이메일을 송, 수신할 계정이 필요하다. 본 연구에서는 프로그램의 구현 가능성만을 확인하기 위해 송, 수신 계정을 동일시 하였으며 Google의 앱 비밀번호 기능을 사용하기 위해서 Gmail 계정을 사용하였다.

먼저"account.py"파일을 생성한 뒤 파일을 송신할 이메일 주소와 그것의 앱 비밀번호를 입력하였고 다시 처음 파일로 돌아와 smtplib을 import 해주었다. 이메일 본문에 연구1에서 저장했던 변수인 Info를 입력했으며 이메일의 제목은 최상단 공지의 제목으로 설정하였다.

라. 연구 결과

○ 연구1과 연구2를 통해 구현한 프로그램을 실행시키면 Chrome 브라우저가 실행되고 약 13~51초 사이에 웹 크롤링 및 이메일 전송 작업을 수행한다. 프로그램은 공지 정보를 수집해 텍스트를 가져온 뒤 미리 설정해 놓은 이메일 주소로 그것을 전송한다.

Fig. 3. 이메일에 삽입된 요소들의 웹 페이지 상의 위치

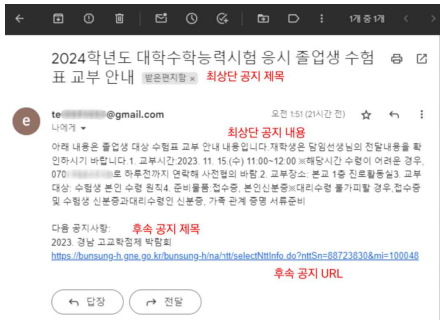

Fig. 4. 프로그램 실행을 통해 전송된 이메일

3. 결 론

가. 결과 해석

○ Selenium을 활용하여 웹 페이지를 자동 탐색하고 정보를 수집하며, 수집한 정보를 이메일을 통해 자동으로 전송하는 기능을 구현했다. 실행 결과로는 Chrome 브라우저가 자동으로 열린 뒤, 웹 크롤링 및 이메일 전송 작업이 약 13~51초 사이에 완료된다는 것을 확인했다. 실행 도중에 Chrome 브라우저를 종료할 경우에는 작업이 완료되지 않는다.

나. 시사점

○ 본 연구에서는 Python 프레임워크를 통해 웹 정보 수집 및 처리를 거친 뒤 이메일 전송을 하는 프로그램을 개발함으로써 웹 페이지 정보의 선택적 수집 및 이메일을 통한 송, 수신의 자동화가 가능함을 시사한다. 본 연구의 개선점은 크게 3가지로 구분할 수 있다.

- 프로그램이 작업을 수행하는 시간이 약 13~51초 사이로 사용자의 인터넷 환경에 따라 작업 소요 시간의 폭이 크게 차이가 난다.
- 직접 실행시켜야지만 자동으로 웹 정보를 수집한다는 점에 있어서 완전한 자동이라고 보기 어렵다.
- 첨부파일이 있을 시 직접 페이지에 접속해 내려받아 주어야 한다.

따라서 후속 연구는 실시간으로 학교 홈페이지 정보를 업데이트해 새로운 공지가 업로드 되면 그 순간에 프로그램을 실행하는 방식으로 진행돼야 할 것이다. 또 이번 연구에서 실현되지 않았던 기능인 첨부파일 탑재도 가능할 수 있게끔 연구를 진행해야 할 것이며 인터넷 환경에 따른 프로그램 동작시간 문제도 해결해야 할 것이다.

4. 참 고 문 헌

- 이정환. 2016,“공학적데이터처리를위한파이썬(Python)언어의활용”, 전기의세계,65,541-48.

- 서동민 and 정한민. (2013). 빅데이터 분석 서비스 지원을 위한 지능형 웹 크롤러. 한국콘텐츠학회 논문지, 13(12), 575-584.

- Na, C.-W., & On, B.-W. (2019). A proposal on a proactive crawling approach with analysis of state-of-the-art web crawling algorithms. *Journal of Internet Computing and Services*, *20*(3), 43-59. https://doi.org/10.7472/JKSII.2019.20.3.43

- https://www.selenium.dev/about/

- https://en.wikipedia.org/wiki/Simple_Mail_Transfer_Protocol

- https://docs.python.org/ko/3/library/smtplib.html