

Weather Info

Python Console Weather Info project

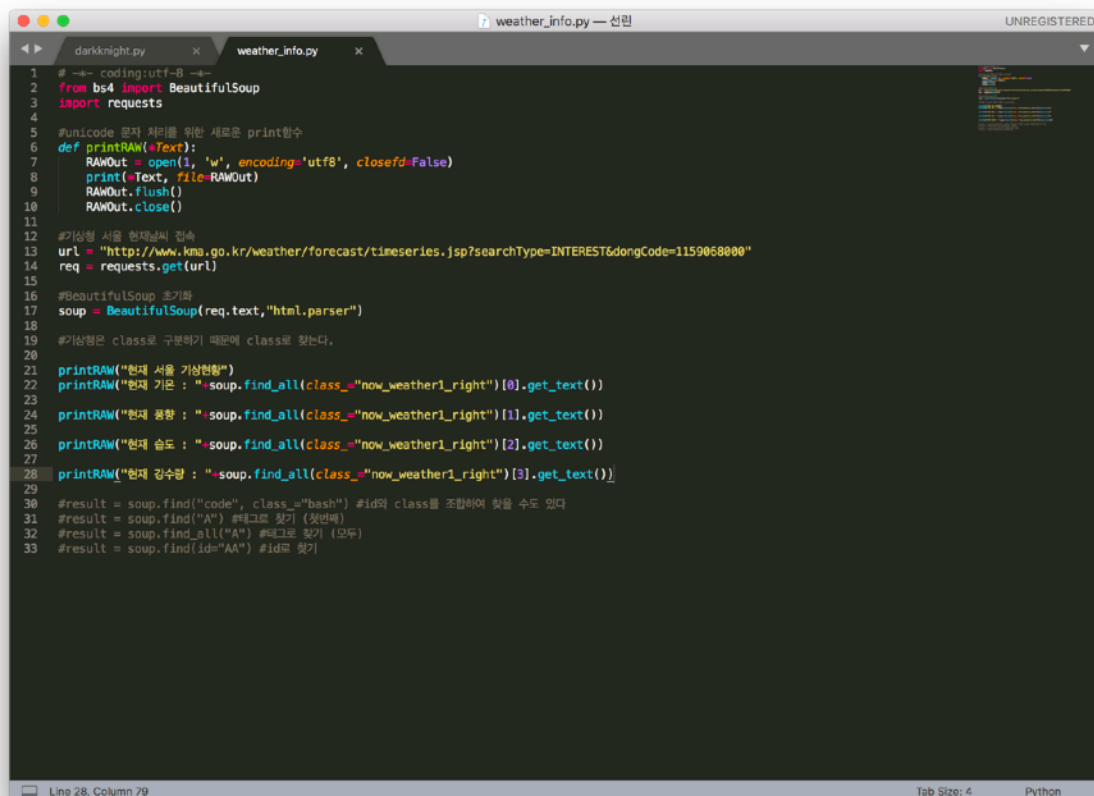
i. 프로젝트 개요

터미널 환경에서 작업할 일이 많은 선린인터넷고등학교의 학생들은 터미널, 콘솔 환경에서 실시간 날씨 정보를 알 수 있는 명령어가 없기 때문에 날씨를 알고 싶으면 번거롭게 외부 프로그램을 실행하거나 인터넷에 접속해야 합니다.

따라서 우리는 터미널에서 작업하는 도중에도 명령어 한 번으로 날씨 정보를 한눈에 확인할 수 있는 프로그램을 개발했습니다. 그것은 바로 weather_info 프로그램입니다.

ii. 소스코드 설명

소스코드는 아래와 같습니다.



```
1 # -*- coding: utf-8 -*-
2 from bs4 import BeautifulSoup
3 import requests
4
5 #unicode 문자 처리를 위한 새로운 print함수
6 def printRAW(sText):
7     RAWOut = open(1, 'w', encoding='utf8', closefd=False)
8     print(sText, file=RAWOut)
9     RAWOut.flush()
10    RAWOut.close()
11
12 #기상청 서울 현재날씨 접속
13 url = "http://www.kma.go.kr/weather/forecast/timeseries.jsp?searchType=INTEREST&dongCode=1159068000"
14 req = requests.get(url)
15
16 #BeautifulSoup 초기화
17 soup = BeautifulSoup(req.text, "html.parser")
18
19 #기상청은 class로 구분하기 때문에 class로 찾는다.
20
21 printRAW("현재 서울 기상현황")
22 printRAW("현재 기온 : " + soup.find_all(class_="now_weather1_right")[0].get_text())
23
24 printRAW("현재 풍향 : " + soup.find_all(class_="now_weather1_right")[1].get_text())
25
26 printRAW("현재 습도 : " + soup.find_all(class_="now_weather1_right")[2].get_text())
27
28 printRAW("현재 강수량 : " + soup.find_all(class_="now_weather1_right")[3].get_text())
29
30 #result = soup.find("code", class_="bash") #id와 class를 조합하여 찾을 수도 있다
31 #result = soup.find("A") #태그로 찾기 (첫번째)
32 #result = soup.find_all("A") #태그로 찾기 (모두)
33 #result = soup.find(id="AA") #id로 찾기
```

소스코드는 Python답게 매우 단순합니다. 기상청 웹 사이트에 접속해서 날씨 정보를 파싱해 오는 것이기 때문에 파싱을 위한 BeautifulSoup 패키지와 접속을 위한 requests 라이브러리를 사용했습니다.

또한, 소스코드의 상단을 보면 일반적인 Python의 print 함수를 재정의 한 것을 볼 수 있습니다. 저것은 unicode 형식의 문자를 출력할 수 없는 기존의 print함수를 unicode도 출력할 수 있게 만든 것입니다. 처음에 기상청 웹 사이트에서 파싱을 해 왔을 때, print 함수로 출력할 수 없는 기호가 포함되어 있었기 때문에 정상적으로 프로그램이 구동되지 않았습니다. 따라서 구글링을 한 결과, github에 관련 게시글이 올라와 있었고, 그 게시글을 참고하여 문제를 해결했습니다.

그리하여 문제를 해결한 후, 우리는 서울의 기온, 풍향, 습도, 강수량을 실시간으로 파싱해 출력해주는 부분을 구현했습니다. 파싱을 해오는 방법은 여러 가지가 있는데,

```
▶ <dd class="now_weather1_left icon">...</dd>
  <dd class="now_weather1_right temp1 MB10">-1℃</dd>
  <dd class="now_weather1_right">남동 0m/s</dd>
  <dd class="now_weather1_right">74%</dd>
  <dd class="now_weather1_right">--</dd>
```

기상청은 class라는 값으로 날씨정보를 구분해주고 있기 때문에 `soup.find_all(class_="now_weather1_right")` 라는 코드를 통해 값을 가지고 왔습니다.

iii. 프로젝트 점검 및 개선

프로그램 구현이 다 되었을 때, 우리는 우리의 최종 목표가 무엇인지 다시 한 번 생각해 보았습니다. 우리가 추구하는 가치는 ‘편리함’ 이었습니다. 하지만 환경변수에 프로그램이 등록되어 있지 않았을 때 프로그램의 경로를 모두 작성해줘야 했기 때문에 상당히 번거롭고 편리하지 않았습니다. 따라서 우리는 어떤 경로에서 작업을 하던 명령어 한 단어로 프로그램을 실행시키기 위해 환경변수를 만들어 주었습니다.

아래 사진과 같이 사용자의 홈 디렉토리에 있는 .bash_profile 파일에 환경변수 설정을 추가해 주었습니다. 저 파일에 환경변수 설정을 해 주면 컴퓨터를 다시 시작해도 환경변수 설정이 남아있어 언제든지 사용할 수 있습니다. 환경변수 이름은 weather로 했습니다. 이제, 쉘의 어디서든 환경변수를 통하여 날씨정보를 알아올 수 있습니다.

```
snow — vi ./bash_profile — 90x24

# Setting PATH for Python 3.6
# The original version is saved in .bash_profile.pysave

PATH="/Library/Frameworks/Python.framework/Versions/3.6/bin:${PATH}"

export weather="python3 /Users/snow/Desktop/선 린 /UniFox_Python_Project/weather_info.py"
export PATH=${PATH}:${weather}/bin
~
~
~
~
~
~
~
~
~
~
~
```

```
snow — -bash — 90x24

baek-chang-in-ui-MacBook-Pro:~ snow$ $weather
현재 서울 기상 현황
현재 기온 : 2℃
현재 풍향 : 남 동 1m/s
현재 습도 : 73%
현재 강수량 : -
baek-chang-in-ui-MacBook-Pro:~ snow$
```

위와 같이 \$weather 명령만 명령줄에 입력하면 언제든지 날씨 정보를 가져올 수 있습니다. 드디어 최종적으로 우리가 원하던 기능을 구현했습니다.

iv. 요약 및 미래

지금까지 여러 프로그램 개발 프로젝트를 하면서 실생활에서 자주 사용하게 되는 프로그램을 만들어본 적이 거의 없었는데, 이번 프로젝트는 평소에 정말 필요했던 프로그램을 개발한 것이기 때문에 더욱 그 의미가 깊다고 생각합니다.

우리의 프로젝트는 여기서 끝났지만, 버전 업데이트와 유지보수는 여기서 끝이 아닙니다. 앞으로 사용자의 위치를 고려해 해당 지역의 날씨를 가져오는 기능과 맑음, 흐림 등 날씨를 알 수 있는 기능과 현재 시간을 출력해주는 기능 등을 추가하여 실제 사용할 계획입니다.