



PYTHON을 이용한 웹크롤링 분석

Web 기반 인공지능
개발자 과정 교육생
-장호성-

목 차

◆ 분석 도구

◆ 분석 동기

◆ 코드 분석

-응용

분석 도구



Visual studio code



python

Python



Excel



Chrome

분석 주제 및 동기

주제 : 국민연금 지급예상액 분석

동기 : 국민 연금에 가입하고 일정기간 이상 납부하여 60세 이상이면 노령연금을 지급받는데 입금한 연금 금액과 입금 연차별로 받는 월 연금액수가 달라 알아보기가 힘들다 . 그래서 알아보기 쉽게 웹페이지에 있는 자료를 파이썬으로 웹크롤링하여 여러가지 방식으로 분석하게 되었다.

분석 참고 자료

예상연금 간단계산 - Chrome

nps.or.kr/jsppage/app/etc/simpleExpect.jsp

예상연금 간단계산

월 납입보험료 원

고객님 월 납입보험료에 해당하는 소득기준은 원이며,
지금 적용되는 "A"값(전체 가입자 소득평균액)을 기준으로 산출된 예상연금액은 아래와 같습니다.
국민연금 콜센터(국번없이 1355)로 전화하시면 상세내역(11년 가입하 경우의 같은 5년 단위 이하의
지급예상액 등)을 상담받을 수 있습니다.

노령연금 (매월 지급예상액)

10년 가입	15년 가입	20년 가입	25년 가입	30년 가입	35년 가입	40년 가입
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>

장애연금 (매월 지급예상액)

장애 1급	장애 2급	장애 3급	장애 4급(일시금)
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>

유족연금 (매월 지급예상액)

국민연금공단
예상연금 간단계산

[https://www.nps.or.kr/jsppage/
app/etc/simpleExpect.jsp](https://www.nps.or.kr/jsppage/app/etc/simpleExpect.jsp)

웹 크롤링 코드

```
from bs4 import BeautifulSoup
import requests
from selenium import webdriver
```

```
insu = 100000
url = "https://www.rps.or.kr/rpsage/app/etc/singleObject.do?yyyy=2022&status=calendar-471532&min=2330061151-{insu}&inquiry=202315300873&Sv=31300931907+90.986"
req = requests.get(url)
req.encoding = "utf-8"
html = BeautifulSoup(req.content, "html.parser")
data_1 = html.select("span.data")
```

```
years10 = data_1[0].string
years15 = data_1[1].string
years20 = data_1[2].string
years25 = data_1[3].string
years30 = data_1[4].string
years35 = data_1[5].string
years40 = data_1[6].string
```

```
a = [years10, years15, years20, years25, years30, years35, years40]
print("=====>{a}")
```

웹 크롤링 상세

라이브러리

```
from bs4 import BeautifulSoup
import requests
```

BeautifulSoup : 웹페이지의 정보를 쉽게 스크랩할 수 있도록 사용

requests : 파이썬에서 HTTP를 호출할때 사용
ex) request.get()

URL 및 HTTP 호출

```
insu = 90000
url = f'https://www.insu.co.kr/jspage/app/abc/sampleExpect.do?sp=177/2929&status=call&acc=47169065701=29790621194=-17191181100017=3817599883070437540891327580827458419363089803075014315063'
req = requests.get(url)
req.encoding = "utf-8"
html = BeautifulSoup(req.content, "html.parser")
```

insu는 소득기준 100만원을 기준 90000

웹 크롤링 상세

연도별 연금값 찾기

```
data_1 = html.select("span.data")
years10 = data_1[0].string
years15 = data_1[1].string
years20 = data_1[2].string
years25 = data_1[3].string
years30 = data_1[4].string
years35 = data_1[5].string
years40 = data_1[6].string
```

select로 웹크롤링할때 원하는페이지에서 웹페이지 검사로 span.data에서 원하는 수치를 얻기위해서 span.data에서 크롤링하였다. string은 불필요한 태그가 나오는것을 방지하기 위해서 넣었다.

웹 크롤링 상세

연금값 출력

```
a = [years10, years15, years20, years25, years30, years35, years40]  
print(f"노형연금은{a}")
```

연금값 print

```
"c:/Users/User/Desktop/프로젝트1/새 폴더/새 폴더/장호성/case.py"  
노형연금은['325,920', '484,710', '643,500', '802,300', '961,090', '1,119,880', '1,278,680']
```

웹 크롤링 응용

- ◆ 엑셀파일로 결과 출력

- ◆ Flask 활용 Chrom환경에서 출력

웹 크롤링 응용

엑셀 파일로 결과 출력

라이브러리 openpyxl 사용

```
from openpyxl import Workbook
wb = Workbook()
ws = wb.active

ws.append([f"노령연금 임금금액 = {insu}"])
ws.append(["10년", "15년", "20년", "25년", "30년", "35년", "40년"])
ws['A3'] = years10
ws['B3'] = years15
ws['C3'] = years20
ws['D3'] = years25
ws['E3'] = years30
ws['F3'] = years35
ws['G3'] = years40

wb.save('yungum.xlsx')
wb.close()
```

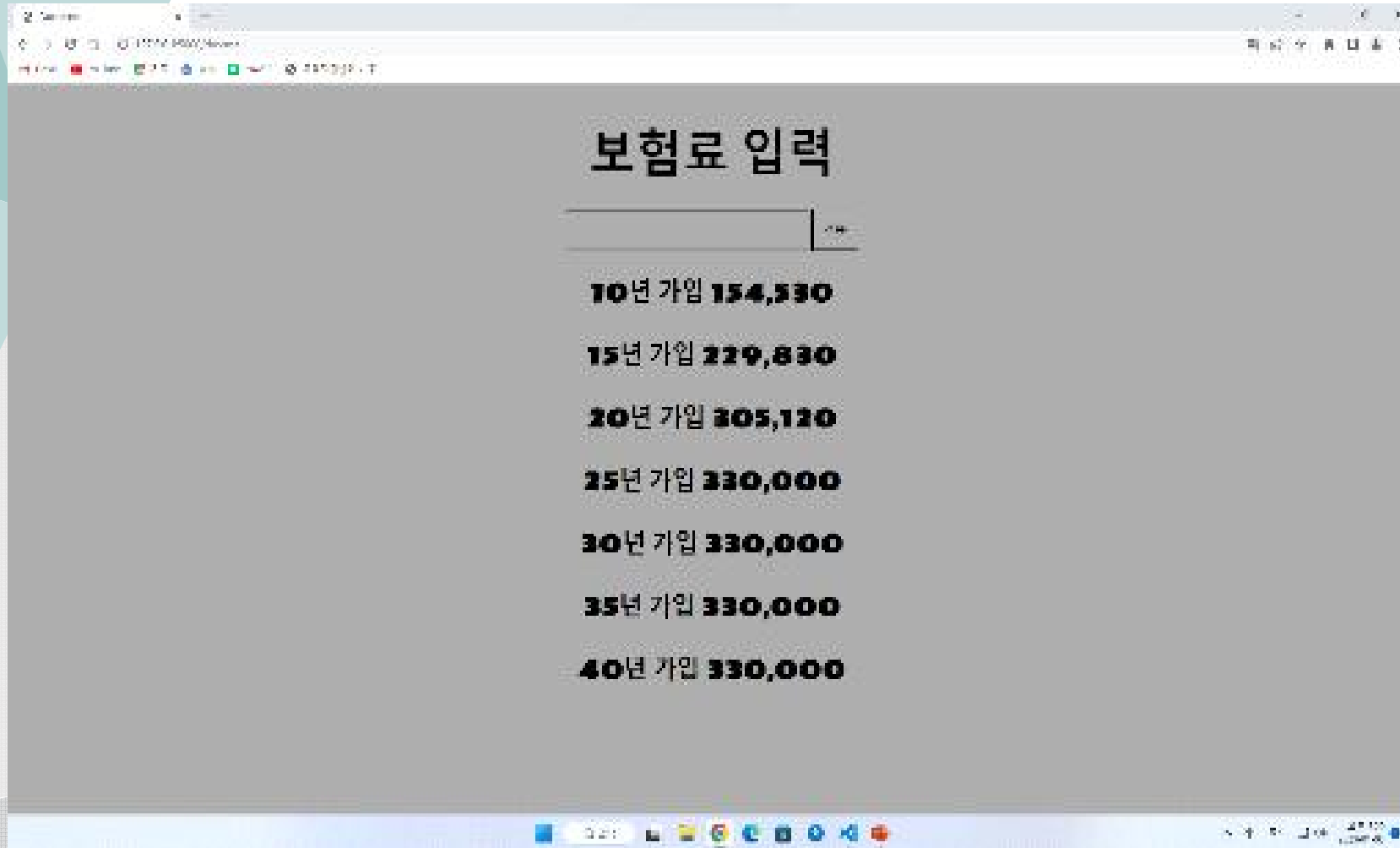
웹 크롤링 응용

엑셀 출력 결과

노령연금 입금금액 = 330000						
10년	15년	20년	25년	30년	35년	40년
325,920	484,710	643,500	802,300	961,090	1,119,880	1,278,680

웹 크롤링 응용

Flask 활용 Chrom환경에서 출력



보험료 입력

10년 가입	154,530
15년 가입	229,830
20년 가입	305,120
25년 가입	330,000
30년 가입	330,000
35년 가입	330,000
40년 가입	330,000

보험료 입력시
자동으로 연금계산

웹 크롤링 응용

자료 가져오기

```
from bs4 import BeautifulSoup
import requests
```

def 사용

{insu}에 보험료 입력

[illegible]

웹 크롤링 응용

플라스크 라이브러리를 활용

```
from flask import Flask, render_template, request
import dusrma as ds

app = Flask(__name__)

@app.route("/")
def hello():
    return render_template('hello.html')

@app.route("/dusrma", methods=['POST', 'GET'])
def dusrma():
    insu = 330000 #초기값으로 초기화
    if request.method == 'POST': #폼태그 형식이 post
        insu = request.form['won']
    result = ds.dusrma(insu)
    return render_template("dusrma.html", result=result)

if __name__ == '__main__':
    app.run(debug=True)
```

←def dusrma을 ds로 지정

←form 생성

웹 크롤링 응용

Chrom환경에서 출력

```
<!doctype html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport"
    content="width=device width, user-scalable=no, initial scale=1.0, maximum scale=1.0, minimum scale=1.0">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">
  <title>Document</title>
  <style>
    <import url('https://fonts.googleapis.com/css2?family=SeymourOne&display=swap')>
    {
      font-family: 'Seymour One', sans-serif; /* 없으면 sans-serif 사용 */;
      font-style: inherit;
      font-weight: bold;
      padding: 0;
      margin: 0;
      box-sizing: border-box;
      background-color: # darkgray;
    }
    body{
      text-align: center;
      margin: 30px;
      line-height: 50px;
    }
  </style>
</head>
<body>
  <font style="font size:40px">보험료 입력</font>
  <form method="post">
    <input type="text" name="won" id="" style="padding: 15px;font size:15px;border: radius 7px;" >
    <input type="submit" style="padding: 15px;font-size: 15px;" value="전송"/>
  </form>
  <div>10년 가입 <{{result[0]}}</div>
  <div>15년 가입 <{{result[1]}}</div>
  <div>20년 가입 <{{result[2]}}</div>
  <div>25년 가입 <{{result[3]}}</div>
  <div>30년 가입 <{{result[4]}}</div>
  <div>35년 가입 <{{result[5]}}</div>
  <div>40년 가입 <{{result[6]}}</div>
</body>
</html>
```

←보험료 입력 form

보험료 입력

164000

월

10년 가입 230,990

15년 가입 343,530


20년 가입 456,080

25년 가입 568,620

30년 가입 681,160

35년 가입 793,710

40년 가입 906,250



감사합니다