■ 서식 2 : 결과보고서 요약본

Project Based Learning 결과보고서 요약본

<table>
<tr><td rowspan="2">교과목명</td><td>국문</td><td colspan="3">인공지능과 데이터과학</td></tr>
<tr><td>영문</td><td colspan="3">Artificial Intelligence and Data Science</td></tr>
<tr><td colspan="2">PBL 관련 능력단위
(능력단위코드)</td><td colspan="3">PBL 관련 능력단위요소 (능력단위요소코드)</td></tr>
<tr><td colspan="2" rowspan="2">웹 사이트 크롤링하기</td><td colspan="3">웹 사이트 크롤링하기 / 설계 및 분석하기</td></tr>
<tr><td colspan="3">프로젝트 계획서 작성하기</td></tr>
<tr><td colspan="2">학년 반</td><td>2학년 1반</td><td>조원</td><td>백승원, 이민욱,
이은우, 우재윤, 강대현</td></tr>
<tr><td colspan="2">프로젝트 주제</td><td colspan="3">이디야 커피 웹사이트 정적 크롤링</td></tr>
<tr><td colspan="2">지도교수</td><td>금 득 규</td><td>산업체 참가여부</td><td>☐ 유 ☑ 무</td></tr>
<tr><td colspan="2">참여학생</td><td colspan="3">백승원, 이민욱, 이은우, 우재윤, 강대현</td></tr>
<tr><td colspan="2">작품 개요
(주제 선정 이유)</td><td colspan="3">팀원들과의 토의를 통해 프로젝트를 진행할 크롤링 방식과 크롤링할 커피 브랜드를 선정하였습니다.

토의 결과 정적 크롤링이 가장 원활하게 프로젝트를 진행할 수 있는
크롤링 방식이라는 결론이 나서 정적 크롤링으로 프로젝트를 진행하였습니다.

선정한 커피 브랜드인 이디야 커피는 다수결 투표를 통해서 정하였습니다.</td></tr>
<tr><td colspan="2">작품 구조도
(문제점 제시 및 개선방안)</td><td colspan="3">프로젝트 전체 과정은 파이썬 코드로 이디야 커피 웹 사이트를
정적 크롤링한 후에 크롤링된 데이터를 분석하는 과정으로
진행되었습니다.</td></tr>
<tr><td colspan="2">관련 이론</td><td colspan="3">웹 클라이언트와 웹 서버 사이의 HTTP 통신 원리.
파이썬 프레임워크를 사용한 크롤링 수행 원리.</td></tr>
<tr><td colspan="2">결과물 제작
(문제점 개선사항)</td><td colspan="3">커피브랜드 이디야 웹페이지 정적 크롤링 코드

```
import urllib.request
from bs4 import BeautifulSoup
import pandas as pd
def Ediya_menu(result):
    Ediya_url = 'https://ediya.com/contents/drink.html'
    html = urllib.request.urlopen(Ediya_url)
    soupEdiya = BeautifulSoup(html, 'html.parser')
    menu_items = soupEdiya.find_all('div', class_='pro_detail')
    for menu in menu_items:
        if menu:
            menu_name = menu.find('h2').text.strip()
            menu_detail = menu.find('p').text.strip()
      menu_nutri = menu.find('div', class_='pro_nutri').text.strip()
      menu_allergy = menu.find('div', class_='pro_allergy').text.strip()
      result.append([menu_name, menu_detail, menu_nutri, menu_allergy])
def main():
     result = []
```

</td></tr>
</table>

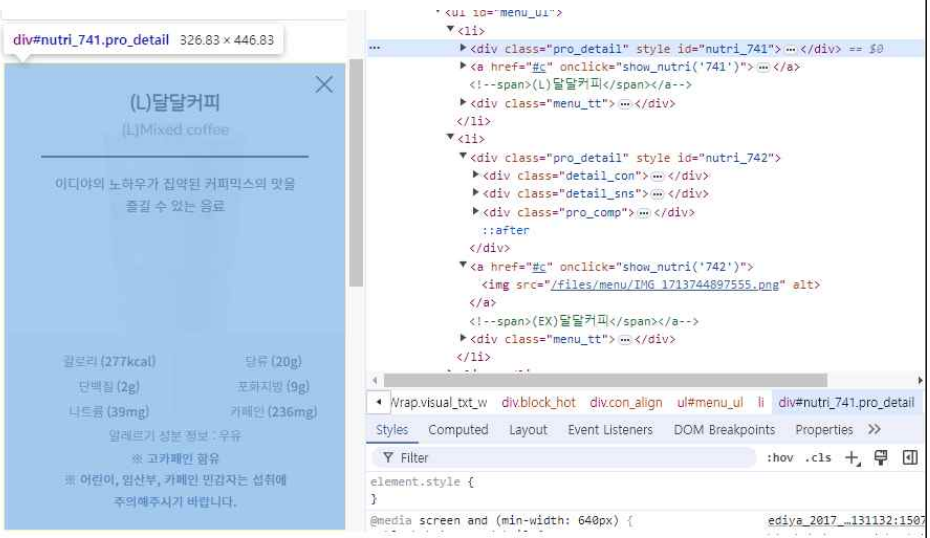
```
    Ediya_menu(result)
    Ediya_tbl = pd.DataFrame(result, columns=('name', 'detail', 'nutri', 'allergy'))
    Ediya_tbl.to_csv('Ediya_menu.csv', encoding='utf-8-sig', mode='w', index=False)
if __name__== '__main__':
main()
```

해당 코드는 크롤링을 위한 BeautifulSoup라이브러리와 서버에 url을 요청해 html을 받기 위한 urllib.request라이브러리, 그리고 추출한 결과인 데이터를 분석 하기위한 pandas라이브러리를 이용한다.

먼저 Ediya_menu()라는 함수를 만드는데, 여기서는 웹 API를 이용한 HTTP 요청과 응답이 오고간다. 첫째로 html을 가져올 url을 입력하고, 둘째론 그 url에서 받은 html을 저장하고, 셋째엔 BeautifulSoup객체를 생성하고, 네번째로는 필요한 부분의 태그와 클래스를 분석해 파싱해서 menu_item으로 저장한다.

그렇게 menu_items에 저장된 값들중 h2태그에 해당하는 부분[음료 이름]을 menu_name에, p에 해당하는 부분[음료 설명]을 menu_detail, div 태그에 해당하고, pro_nutri 클래스에 해당 부분[음료 영양분]을 menu_nutri, div 태그에 해당하고, pro_allergy 클래스에 해당하는 부분[음료 알러지]을 menu_allergy에 저장하게 된다.

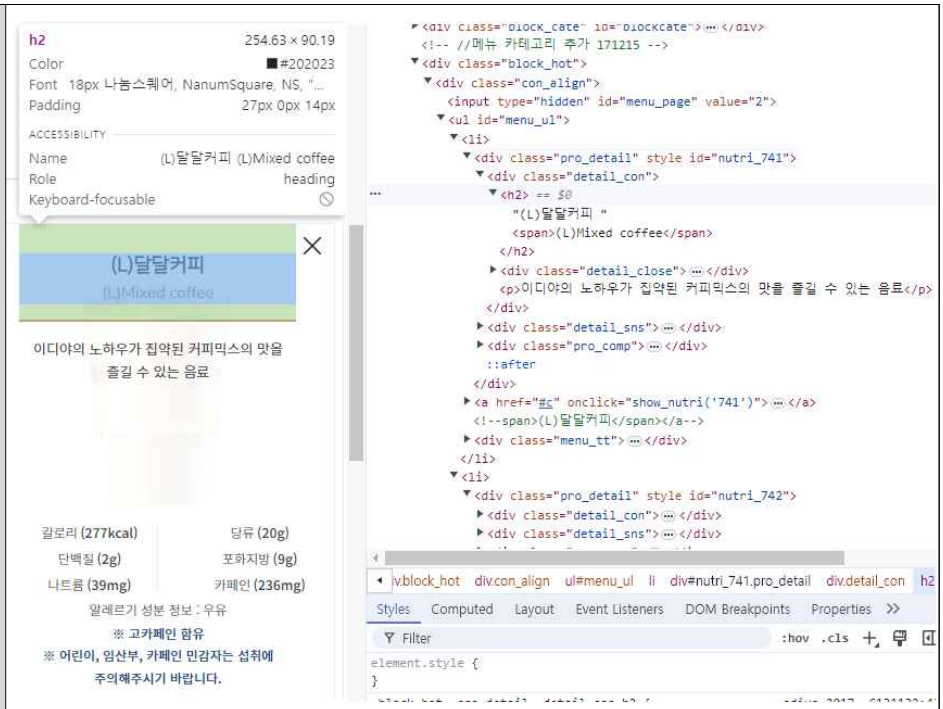
마지막으론 main()함수이며 추출한 결과를 저장할 공간을 생성하고, Ediya_menu()함수를 호출한다. 이때 위에서 설명한 Ediya_menu()함수의 동작과정이 수행된다. 그렇게 추출한 결과를 판다스의 pd.DataFrame()기능을 이용해 데이터 프레임으로 저장하고, 저장한 값들로 csv파일을 생성한다.



이 사진은 음료 정보에 해당하는 HTML 코드이며 보다시피 태그가 div로 시작되고 클래스는 "pro_detail"로 되어있어 " <div class= 전체적인 음료정보가 담겨있는 틀이라고 볼 수 있다.

```
menu_items = soupEdiya.find_all('div', class_='pro_detail')
```

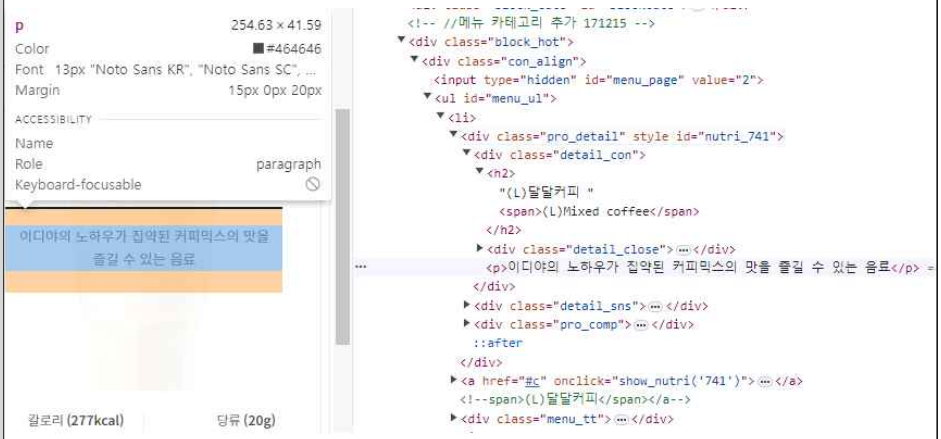
즉 해당 코드는 그 전체적인 음료정보 틀의 첫번째 줄부터 끝맺음에 해당하는 부분까지 파싱을 한다는 코드이다.



이 사진은 음료 이름에 해당하는 부분이며 태그가 h2로 시작한다는걸 확인할 수 있다.

```
menu_name = menu.find('h2').text.strip()
```

그러므로 다음과 같이 코드를 짜서 menu_name변수에 음료 이름을 저장을 하고,



이건 그 음료의 설명에 해당하는 부분이며 태그가 p로 시작함을 알 수 있다.

```
menu_detail = menu.find('p').text.strip()
```

그러므로 이것도 다음과 같이 코드를 짜고 menu_detail에 저장한다.

이건 음료의 영양 함류량을 표현하는 태그이며 div태그로 시작하고 클래스의 이름이 "pro_nutri로 되어있음을 나타낸다.

```
menu_nutri = menu.find('div', class_='pro_nutri').text.strip()
```

그렇기에 다음과 같이 파싱하는 코드를 짜주었다.

마지막으로는 음료에 알러지 성분을 나타내는 태그로 div태그와 "pro_allergy"라는 클래스 이름을 사용하였다.

```
menu_allergy = menu.find('div', class_='pro_allergy').text.strip()
result.append([menu_name, menu_detail, menu_nutri, menu_allergy])
```

마지막으로는 이렇게 추출한 정보들을 result라는 변수에 저장해준다.

Ediya_menu.csv

1 to 9 of 9 entries Filter

name	detail	nutri	allergy
카페 아메리카노 Caffe Americano	이디야의 스모키한 맛과 풍부한 바디감을 느낄 수 있는 이디야 대표 음료	칼로리 (12kal) 포화지방 (12kal) 당류 (0g) 나트륨 (2mg) 단백질 (20g) 카페인 (22mg)	알레르기 유발요인: 계란, 우유, 대두, 밀, 쇠고기, 닭고기 함유
(L)달달커피 (L)Mixed coffee	이디야의 노하우가 집약된 커피믹스의 맛을 즐길 수 있는 음료	칼로리 (277kcal) 당류 (20g) 단백질 (2g) 포화지방 (9g) 나트륨 (39mg) 카페인 (236mg)	알레르기 성분 정보 : 우유
(EX)달달커피 (EX)Mixed coffee	이디야의 노하우가 집약된 커피믹스의 맛을 즐길 수 있는 음료	칼로리 (323kcal) 당류 (23g) 단백질 (2g) 포화지방 (11g) 나트륨 (46mg) 카페인 (276mg)	알레르기 성분 정보 : 우유
팥인절미 1인빙수 Red Bean Injeolmi	아삭한 얼음과 팥이 어우러진 베이스에 아이스크림, 통팥, 인절미,시리얼을 올린 빙수	칼로리 (569kcal) 당류 (53g) 단백질 (11g) 포화지방 (6g) 나트륨 (218mg) 카페인 (0mg)	알레르기 성분 정보 : 우유, 대두
망고 요거놀라 1인빙수 Mango Yogurt Granola	아삭한 얼음과 요거트가 어우러진 베이스에 아이스크림, 망고패션베이스, 그래놀라를 올린 빙수	칼로리 (452kcal) 당류 (56g) 단백질 (7g) 포화지방 (7g) 나트륨 (93mg) 카페인 (0mg)	알레르기 성분 정보 : 우유, 대두
초당 옥수수 1인 빙수 Sweet Corn	알알이 씹히는 고소한 옥수수수와 바삭한 콘후레이크의 식감이 어우러진 여름시즌 한정 초당 옥수수 빙수	칼로리 (396kcal) 당류 (41g) 단백질 (8g) 포화지방 (7g) 나트륨 (131mg) 카페인 (0mg)	알레르기 성분 정보 : 우유, 대두
팥인절미 눈꽃빙수 Red Bean Injeolmi	부드러운 우유 눈꽃 빙수에 국산팥과 인절미를 올려 고소하고 달콤하게 즐길 수 있는 빙수	칼로리 (771kcal) 당류 (52g) 단백질 (18g) 포화지방 (6g) 나트륨 (331mg) 카페인 (0mg)	알레르기 성분 정보 : 우유, 대두, 밀
애플망고 눈꽃빙수 Apple Mango	부드러운 우유 눈꽃 빙수에 달콤한 망고를 더해 시원하고 달콤하게 즐길 수 있는 빙수	칼로리 (533kcal) 당류 (102g) 단백질 (7g) 포화지방 (5g) 나트륨 (97mg) 카페인 (0mg)	알레르기 성분 정보 : 우유, 대두
이디야 콤부차 복숭아 망고 Peach & Mango	홍차와 녹차의 추출액과 프락토올리고당, 스코비로 발효한 콤부차 원액과 청제수, 이디야의 노하우가 담긴 과일 티 농축액을 블렌딩하여 만든 프리바이오틱스를 함유한 병음료 콤부차	칼로리 (142kcal) 당류 (32g) 단백질 (0g) 포화지방 (0g) 나트륨 (0mg) 카페인 (0mg)	알레르기 성분 정보 : 복숭아

```
Ediya_tbl = pd.DataFrame(result, columns=('name', 'detail', 'nutri', 'allergy')) #추출한 결과를 데이터프레임으로 저장
Ediya_tbl.to_csv('Ediya_menu.csv', encoding='utf-8-sig', mode='w', index=False) # Ediya_menu.csv파일로 저장
```

그러고는 마지막으로 result에 저장된 정보들을 판다스를 이용하여 테이블 형태의 데이터 프레임으로 저장을 하고 csv파일로 저장하여 위 사진과 같이 csv파일 결과가 나오게 된다.

기대 효과 및 활용방안

이디야 커피의 웹 사이트를 정적 크롤링함으로써,
데이터 분석 수행 및 모델 학습에 필요한 데이터를 수집하고 활용할 수 있습니다.

2024 년 6 월 14 일

지도교수 금 득 규 (인)