Lab#3, NLP@CGU Spring 2023

This is due on 2023/03/20 16:00, commit to your github as a PDF (lab3.pdf) (File>Print>Save as PDF).

IMPORTANT: After copying this notebook to your Google Drive, please paste a link to it below. To get a publicly-accessible link, hit the *Share* button at the top right, then click "Get shareable link" and copy over the result. If you fail to do this, you will receive no credit for this lab!

LINK: paste your link here

https://colab.research.google.com/drive/1_wbqb-u51jL_EAuai0oeojHwlZpmF8Ke?usp=share_link

---

Student ID:B0928031

Name:鄭茹云

Question 1 (100 points)

Implementing Yahoo Movies Crawler.

1. Design a Yahoo! Movie Crawler.
2. Crawl all the movie information listed in movie_intheaters page
3. The more movie data crawled, the higher the score

按兩下 (或按 Enter 鍵) 即可編輯

```
import  requests
import  re
from  bs4  import  BeautifulSoup


Y_MOVIE_URL_TEMPLATE  =  "https://movies.yahoo.com.tw/movie_intheaters.html?page={}"

#  YOUR  CODE  HERE!
#  IMPLEMENTIG  YAHOO  MOVIES  CRAWLER
```

```
class MovieCrawler(object):

    def __init__(self):
        self.headers = {
            'User-Agent': 'Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit
        self.movies = []

    def get_movies(self, page_url):
        res = requests.get(page_url, headers=self.headers)
        soup = BeautifulSoup(res.text, 'html.parser')

        movie_list = soup.find('ul', class_='release_list')
        for movie in movie_list.find_all('li'):
            movie_dict = {}
            movie_dict['ch_name'] = movie.find('div', class_='release_movie_name').a
            movie_dict['en_name'] = movie.find('div', class_='release_movie_name').f
            movie_dict['movie_url'] = movie.find('div', class_='release_movie_name')
            movie_dict['release_date'] = movie.find('div', class_='release_movie_tim
            movie_dict['intro'] = movie.find('div', class_='release_text').text.stri
            self.movies.append(movie_dict)

        return self.movies

# # DO NOT MODIFY THE VARIABLES
crawler = MovieCrawler()
movies = []
for i in range(1, 3):
    page_url = Y_MOVIE_URL_TEMPLATE.format(i)
    movies_on_page = crawler.get_movies(page_url)
    movies.extend(movies_on_page)

# # THE RESULTS : AS THE FOLLOWING SECTION
# # {'ch_name', 'en_name', 'movie_url', 'release_date', 'intro'}
print(len(movies))
print(*movies, sep="\n")
```

```
30
{'ch_name': '配樂大師顏尼歐', 'en_name': 'Ennio: The Maestro', 'movie_url': 'https://movies.y
{'ch_name': '熊蓋毒', 'en_name': 'Cocaine Bear', 'movie_url': 'https://movies.yahoo.com.tw/mo
{'ch_name': '若愛重來', 'en_name': 'Marriages', 'movie_url': 'https://movies.yahoo.com.tw/mov
{'ch_name': '無人相信的真相', 'en_name': 'La syndicaliste', 'movie_url': 'https://movies.yaho
{'ch_name': '闇黑對決', 'en_name': "The Devil's Deal", 'movie_url': 'https://movies.yahoo.com
{'ch_name': '噩夢輓歌 4K數位修復版', 'en_name': 'Requiem For A Dream', 'movie_url': 'https://
{'ch_name': '人體動物圖鑑：烏龜的殼其實是肋骨', 'en_name': 'Turtle' s Shell is a Human' s Rib
{'ch_name': '流水落花', 'en_name': 'Lost Love', 'movie_url': 'https://movies.yahoo.com.tw/mov
{'ch_name': '聖蛛', 'en_name': 'Holy Spider', 'movie_url': 'https://movies.yahoo.com.tw/movie
{'ch_name': '沙贊! 眾神之怒', 'en_name': 'Shazam! Fury of the Gods', 'movie_url': 'https://mo
{'ch_name': '配樂大師顏尼歐', 'en_name': 'Ennio: The Maestro', 'movie_url': 'https://movies.y
{'ch_name': '熊蓋毒', 'en_name': 'Cocaine Bear', 'movie_url': 'https://movies.yahoo.com.tw/mo
{'ch_name': '若愛重來', 'en_name': 'Marriages', 'movie_url': 'https://movies.yahoo.com.tw/mov
{'ch_name': '無人相信的真相', 'en_name': 'La syndicaliste', 'movie_url': 'https://movies.yaho
{'ch_name': '闇黑對決', 'en_name': "The Devil's Deal", 'movie_url': 'https://movies.yahoo.com
{'ch_name': '噩夢輓歌 4K數位修復版', 'en_name': 'Requiem For A Dream', 'movie_url': 'https://
```

{'ch_name': '人體動物圖鑑：烏龜的殼其實是肋骨', 'en_name': 'Turtle's Shell is a Human's Rib
{'ch_name': '流水落花', 'en_name': 'Lost Love', 'movie_url': 'https://movies.yahoo.com.tw/mov
{'ch_name': '聖蛛', 'en_name': 'Holy Spider', 'movie_url': 'https://movies.yahoo.com.tw/movie
{'ch_name': '沙贊！眾神之怒', 'en_name': 'Shazam! Fury of the Gods', 'movie_url': 'https://mo
{'ch_name': '夢遊樂園', 'en_name': 'Melody-Go-Round', 'movie_url': 'https://movies.yahoo.com.
{'ch_name': '黑的教育', 'en_name': 'Bad Education', 'movie_url': 'https://movies.yahoo.com.tw
{'ch_name': 'TÁR塔爾', 'en_name': 'Tár', 'movie_url': 'https://movies.yahoo.com.tw/movieinfo_
{'ch_name': '驚聲尖叫6', 'en_name': 'Scream VI', 'movie_url': 'https://movies.yahoo.com.tw/mo
{'ch_name': '怪談比留子 數位修復版', 'en_name': 'Hiruko The Goblin', 'movie_url': 'https://mo
{'ch_name': '天生一對2大電影：再續前緣', 'en_name': 'Love Destiny: The Movie', 'movie_url': '
{'ch_name': '尋找第5味', 'en_name': 'Umami', 'movie_url': 'https://movies.yahoo.com.tw/moviei
{'ch_name': '超完美狗保姆', 'en_name': 'My Puppy', 'movie_url': 'https://movies.yahoo.com.tw/
{'ch_name': '蓋世棋蹟', 'en_name': 'The Royal Game', 'movie_url': 'https://movies.yahoo.com.t
{'ch_name': '斷網', 'en_name': 'Cyberheist', 'movie_url': 'https://movies.yahoo.com.tw/moviei

✓ 1 秒 完成時間：下午3:48