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/1yqAWu_zMBFsF3jQuZWREu3noG3_NBQbo?usp=sharing

Student ID: b0928024

Name:莊靜修

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 = "https://movies.yahoo.com.tw/movie_intheaters.html"

# YOUR CODE HERE!
# IMPLEMENTIG YAHOO MOVIES CRAWLER

class MovieCrawler(object):

    def __init__(self):
```

```
      self.movies = []


    def get_movies(self, page_url):
      try:
        resp = requests.get(page_url)
      except:
        resp = None

      if resp and resp.status_code == 200:
        soup = BeautifulSoup(resp.text, 'html.parser')
        mov_list = soup.find_all("div", class_ = "release_info_text")

        for mov in mov_list:
          temp = {}

          ch_name = mov.find("div", class_ = "release_movie_name").find("a")
          temp["ch_name"] = ch_name.text[19: ]

          en_name = mov.find("div", class_ = "release_movie_name").find("div", cla
          temp["en_name"] = en_name.text[21: ]

          movie_url = ch_name["href"]
          temp["movie_url"] = movie_url

          release_date = mov.find("div", class_ = "release_movie_time")
          t = ""
          for n in release_date.text:
            if (n not in ("\n", " ")):
              t += n
          temp["release_date"] = t

          intro = mov.find("div", class_ = "release_text").find("span")
          temp["intro"] = intro.text

          self.movies.append(temp)

      return self.movies

for page in range(1, 9):

# # DO NOT MODIFY THE VARIABLES
crawler = MovieCrawler()
movies = crawler.get_movies(Y_MOVIE_URL)

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

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