https://drive.google.com/file/d/1mKM9OTGVwofnIKTLjhYCwUpu5uhbrw4A/view?usp=sharing

```
import requests
from bs4 import BeautifulSoup
import json
url = 'https://movies.yahoo.com.tw/moviegenre_result.html?genre_id=19&page=1'
response = requests.get(url=url)
soup = BeautifulSoup(response.text, 'lxml')

info_items = soup.find_all('div', 'release_info')
i=0
link=[0]*10

for item in info_items:
    movie_data = []
    name = item.find('div', 'release_movie_name').a.text.strip()
    english_name = item.find('div', 'en').a.text.strip()

    # 取得電影詳細資訊頁面的網址
    detail_url = item.find('div', 'release_movie_name').a['href']
    detail_resp = requests.get(detail_url)
    detail_soup = BeautifulSoup(detail_resp.text, "html.parser")

    detail_info = detail_soup.find("div", class_="movie_intro_info_r")
    labels = detail_info.find_all("a", class_="gabtn")

    release_time = item.find('div', 'release_movie_time').text.split('：')[-1].strip()
    moive_url = item.find('a',class_="gabtn")['href']
    link[i]=moive_url
    movie_text = item.find('span', 'jq_text_overflow_180 jq_text_overflow_href_list').text

    #intro_element = requests.get(detail_url)
#   intro_soup = BeautifulSoup(intro_element.text, "html.parser")
    #intro_element = intro_soup.find("div", class_="gray_infobox_inner")
    #intro = intro_element.find_all("span")[1].text.strip()
                        # 從 HTML 元素中找到上映日期、簡介等資訊
    #released_date = intro_element.find_all("span")[0].text.strip()
#   intro = intro_element.find_all("span")[1].text.strip()

    links = []
    for label in labels:
        links.append(label["href"].split("/")[-1])


    # 將 labels 轉換為字串
    label_str = ', '.join([label.text.strip() for label in labels])
    print('doc_id : ', i)
    print('cname : ', name)
    print('ename', english_name)
    #print('pagerank', "")
#   print('label[class] : ', {"class": links[1:]})
    print('intro : ', movie_text)
    print('released_date : ', release_time)
    print('links[doc_id] : ', link[i])
    #print('chname:{} enname: ({}) 上映日：{} LABEL: {} '.format(name, english_name, release_time , label_str))
    i+=1


    movie_data.append({
                        "doc_id": i,
                        "cname": name,
                        "ename": english_name,
                        "intro": movie_text,
                        "released_date": release_time,
                        "links": moive_url
                })

    i_str=str(i)
    filename='movie_'+i_str+'.json'
    with open(filename, "w", encoding="utf-8") as f:
        json.dump(movie_data, f, ensure_ascii=False, indent=4)
```

```
doc_id : 0
cname : 2022 TEFF歐洲影展
ename Taiwan European Film Festival
intro :
                一、關於影展
TEFF歐洲影展（Taiwan European Film Festival）自2005年起，於全台各地播放歐洲電影並提供觀眾免費入場觀看，希望台灣民眾藉由欣賞歐洲電影的過程中，

2022年第18屆台灣歐洲影展由歐洲經貿辦事處主辦，外交部、文化部、台北市文化局合辦，與歐盟駐台各代表處協辦，邀請17個歐洲國家個別推選出一部電影參
```

今年影展片單，由歐盟駐台代表高哲夫(Filip Grzegorzeski)處長領軍選片，各歐盟國駐台代表，也由該國選一部最能代表該國的影片，總集成17國、17部影片

綜觀今年17部電影中，有9部是移民、難民議題電影，反映現今歐洲面臨的嚴正課題。另外8部，有溫馨的親情、懸疑的苦戀、清新的愛情、懵懂的青春、與成長

二、影展主題
《邊界.無界》------ 有線的邊界，無界的愛。
在邊界的內外，認同與原鄉、回歸與新生，一再糾結不斷。
在愛的本質上，家庭與親情，男女與愛情，永遠無法割裂。
電影影像與故事文本，如同法國哲學家德勒茲 (Gilles Deleuze) 的空間、時間、影像的辯證關係般。無限循環，永恆回溯，在愛的框內與邊界的框外，一再去

三、影展時間、地點
1、開幕日：2022年11月17日(四) 光點華山電影館
2、全國巡迴放映：2022年11月18日~2023年1月31日，全國各公、私立大專院校、及藝文影視展演空間。

四、主協辦單位
1、主辦單位：歐洲經貿辦事處
2、合辦單位：外交部、文化部、台北市文化局
3、承辦單位：佳映娛樂
4、協辦單位：
奧地利臺北辦事處 Austrian Office, Taipei
比利時臺北辦事處 Belgian Office,Taipei
捷克經濟文化辦事處 Czech Economic and Cultural Office,Taipei
丹麥商務辦事處 Trade Council of Denmark, Taipei
芬蘭商務辦事處 Finland Trade Center in Taiwan
法國在臺協會 French Office in Taipei
德國在臺協會 German Institute in Taipei
匈牙利貿易辦事處 Hungarian Trade Office
義大利經濟貿易文化推廣辦事處 Italian Economic,Trade and Cultural Promotion Office
盧森堡臺北辦事處 Luxembourg Trade and Investment Office
荷蘭貿易暨投資辦事處 Netherlands Trade and Investment Office
波蘭台北辦事處 Polish Office in Taipei
斯洛伐克經濟文化辦事處 Slovak Economic and Cultural Office
西班牙商務辦事處 Spanish Chamber of Commerce
瑞典貿易暨投資委員會台北辦事處 Business Sweden, The Swedish Trade and Invest Council
瑞士商務辦事處 Trade Office of Swiss Industries
英國在台辦事處 British Office
臺灣歐洲聯盟中心 European Union Centre in Taiwan
英國文化協會 British Council
法國文化中心 Institut Français
歌德學院 Goethe Institute, Taipei E España
西班牙教育部 Cervantes Institute
5、協力單位：CinemaWorld 世界電影頻道， Moleskine

五、活動總覽
1、影展期間： 2022年11月18日 (五)至 2023年1月31日(二)
2、放映地點：全國各公、私立大專院校及藝文、影視、展演空間

```
import requests
from bs4 import BeautifulSoup
import json
i=0
link=[0]*10

for genre_id in range(1, 20):

    url = f'https://movies.yahoo.com.tw/moviegenre_result.html?genre_id={genre_id}&page=1'
    r = requests.get(url)
    # 將網頁內容解析為 BeautifulSoup 物件
    soup = BeautifulSoup(r.text, "html.parser")
    last_page = soup.select('ul > li')[-3].text
    last_page1 = int((last_page))
    #print(url)

    for page in range(1, last_page1+1):
        url = f'https://movies.yahoo.com.tw/moviegenre_result.html?genre_id={genre_id}&page={page}'
        #print(url)
        response = requests.get(url=url)
        soup = BeautifulSoup(response.text, 'lxml')

        info_items = soup.find_all('div', 'release_info')
        #print(info_items)

        for item in info_items:
            movie_data = []
            name = item.find('div', 'release_movie_name').a.text.strip()
            english_name = item.find('div', 'en').a.text.strip()

            # 取得電影詳細資訊頁面的網址
```

```
        detail_url = item.find('div', 'release_movie_name').a['href']
        detail_resp = requests.get(detail_url)
        detail_soup = BeautifulSoup(detail_resp.text, "html.parser")

        detail_info = detail_soup.find("div", class_="movie_intro_info_r")
        #labels = detail_info.find_all("a", class_="gabtn")

        release_time = item.find('div', 'release_movie_time').text.split(': ')[-1].strip()
        moive_url = item.find('a',class_="gabtn")['href']
        #link[i]=moive_url
        movie_text = item.find('span', 'jq_text_overflow_180 jq_text_overflow_href_list').text

        #links = []
        #for label in labels:
            #links.append(label["href"].split("/")[-1])


        # 將 labels 轉換為字串
        #label_str = ', '.join([label.text.strip() for label in labels])
        print('doc_id : ', i)
        print('cname : ', name)
        print('ename', english_name)
        #print('pagerank', "")

        print('intro : ', movie_text)
        print('released_date : ', release_time)
        print('links[doc_id] : ', moive_url)



        movie_data.append({
                            "doc_id": i,
                            "cname": name,
                            "ename": english_name,
                            "intro": movie_text,
                            "released_date": release_time,
                            "links": moive_url
                    })

        i_str=str(i)
        filename='movie_'+i_str+'.json'
        with open(filename, "w", encoding="utf-8") as f:
            json.dump(movie_data, f, ensure_ascii=False, indent=4)

        i+=1

#print("ok")
```

這部根據拳擊手回憶錄改編的勵志電影《大獲拳勝》，幸運獲得了奧斯威辛集中營紀念館出借部分場景（該館曾拒絕《辛德勒的名單》拍攝），使得片中對毒氣室

released_date : 2022-08-26
links[doc_id] : https://movies.yahoo.com.tw/movieinfo_main/%E5%A4%A7%E7%8D%B2%E6%8B%B3%E5%8B%9D-the-champion-of-auschwitz-13807
doc_id : 71
cname : 蝙狐俠：英雄覺醒
ename Superwho?
intro :
★法國雙週票房冠軍，賣座突破4億台幣
★法國版《城市獵人》喜劇男星菲力普拉紹自導自演
★《城市獵人》《裸姆Hold不住》製作團隊打造爆笑動作喜劇
★只要有心，人人都能成為超級英雄！

西追（菲力普拉紹 飾）是一名不得志的演員，某天他得到飾演超級英雄「扁糊俠」的機會，原以為可以從此翻身，沒想到一場突如其來的車禍讓他失去記憶，誤

released_date : 2022-08-26
links[doc_id] : https://movies.yahoo.com.tw/movieinfo_main/%E8%9D%99%E7%8B%90%E4%BF%A0-%E8%8B%B1%E9%9B%84%E8%A6%BA%E9%86%92-sup
doc_id : 72
cname : 韓山島海戰
ename Hansan
intro :
★《與神同行》發行團隊年度戰爭動作鉅獻
★ 韓國影史TOP1！觀影人次破1761萬名、票房賣破1357億韓幣《鳴梁：怒海交鋒》精彩前傳震撼登場！
★ 金漢珉導演率領《鳴梁：怒海交鋒》原班團隊斥資300億韓元打造精彩續作！
★《分手的決心》朴海日接棒崔岷植，扮演年輕時期的李舜臣將軍，驍勇應戰！
★ 話題演員 朴海日x卞約漢x安聖基x金成昀x金香起x玉澤演x孔明x朴智煥 華麗共演！
★在鳴梁海戰發生的五年前，改變國家命運的壓倒性勝利之戰即將展開！

1592年4月，朝鮮在壬辰倭亂爆發後，日軍只花了15天就奪下漢陽，讓朝鮮陷入危機中。想一口氣占領朝鮮的日軍，還打算將野心擴展到中國明朝，在釜山浦集結

released_date : 2022-08-19
links[doc_id] : https://movies.yahoo.com.tw/movieinfo_main/%E9%9F%93%E5%B1%B1%E5%B3%B6%E6%B5%B7%E6%88%B0-hansan-13678
doc_id : 73
cname : 子彈列車
ename Bullet Train
intro :
改編自日本暢銷推理小說《瓢蟲》，描述由布萊德彼特領銜主演的殺手「瓢蟲」，在日本高速電鐵的列車上，遇見了許多來自不同背景以及

released_date : 2022-08-03
links[doc_id] : https://movies.yahoo.com.tw/movieinfo_main/%E5%AD%90%E5%BD%88%E5%88%97%E8%BB%8A-bullet-train-12745
doc_id : 74
cname : 殺戮基地
ename Black Site
intro :
★ 《怒火邊界》《捍衛任務》製片打造黑暗系火爆動作鉅片
★ 《不可能的任務系列》動作女星蜜雪兒摩納漢挑大樑主演
★ 《魔鬼終結者：創世契機》男星傑森克拉克演出兇殘反派
★ 揭開正義的表象，裡面往往比你想的更黑暗...

中情局特務艾比（蜜雪兒摩納漢 飾） 在一次恐怖攻擊中失去丈夫和兒子，她自願調往一個專門收留恐怖份子的秘密基地，以調查事故背後的真相。這天，基地

released_date : 2022-07-29
links[doc_id] : https://movies.yahoo.com.tw/movieinfo_main/%E6%AE%BA%E6%88%AE%E5%9F%BA%E5%9C%B0-black-site-13573
doc_id : 75
cname : 外星+人
ename Alienoid

```
# 斷詞
words = jieba.cut(text)

# 將斷詞結果轉換為列表
word_list = list(words)


import requests
from bs4 import BeautifulSoup
import json
i=0
link=[0]*10

for genre_id in range(1, 20):

        url = f'https://movies.yahoo.com.tw/moviegenre_result.html?genre_id={genre_id}&page=1'
        r = requests.get(url)
        # 將網頁內容解析為 BeautifulSoup 物件
        soup = BeautifulSoup(r.text, "html.parser")
        last_page = soup.select('ul > li')[-3].text
        last_page1 = int((last_page))
        #print(url)

        for page in range(1, last_page1+1):
                url = f'https://movies.yahoo.com.tw/moviegenre_result.html?genre_id={genre_id}&page={page}'
                #print(url)
                response = requests.get(url=url)
                soup = BeautifulSoup(response.text, 'lxml')

                info_items = soup.find_all('div', 'release_info')
                #print(info_items)

                for item in info_items:
                        movie_data = []
                        name = item.find('div', 'release_movie_name').a.text.strip()
                        english_name = item.find('div', 'en').a.text.strip()

                        # 取得電影詳細資訊頁面的網址
                        detail_url = item.find('div', 'release_movie_name').a['href']
                        detail_resp = requests.get(detail_url)
                        detail_soup = BeautifulSoup(detail_resp.text, "html.parser")

                        detail_info = detail_soup.find("div", class_="movie_intro_info_r")
                        labels = detail_info.find_all("a", class_="gabtn")

                        release_time = item.find('div', 'release_movie_time').text.split(': ')[-1].strip()
                        moive_url = item.find('a',class_="gabtn")['href']
                        #link[i]=moive_url
                        movie_text = item.find('span', 'jq_text_overflow_180 jq_text_overflow_href_list').text
                        # 斷詞
                        words = jieba.cut(text)

                        # 將斷詞結果轉換為列表
                        word_list = list(movie_text)


                        #links = []
                        #for label in labels:
                                #links.append(label["href"].split("/")[-1])


                        # 將 labels 轉換為字串
                        label_str = ', '.join([label.text.strip() for label in labels])
                        print('doc_id : ', i)
                        print('cname : ', name)
                        print('ename', english_name)
                        #print('pagerank', "")

                        print('intro : ', movie_text)
                        print('released_date : ', release_time)
                        print('links[doc_id] : ',    moive_url)



                        movie_data.append({
                                                "doc_id": i,
                                                "cname": name,
                                                "ename": english_name,
                                                "intro": movie_text,
                                                "released_date": release_time,
                                                "links": moive_url
```

```
                                        })

                        i_str=str(i)
                        filename='movie_'+i_str+'.json'
                        with open(filename, "w", encoding="utf-8") as f:
                                json.dump(movie_data, f, ensure_ascii=False, indent=4)

                        i+=1

#print("ok")


sentences = ['This is the first word',
                    'This is the second text Hello How are you',
                    'This is the third , this is it now']
sentence_dict = {}
index_dict = {}
for index, line in enumerate(sentences):
        line = line.lower()
        sentence_dict[index] = line
        for char in line.split(' '):
                if not char.strip():
                        continue
                if char in index_dict:
                        index_dict[char].add(index)
                else:
                        index_dict[char] = {index}


#檢索
def list_intersection(list1: list, list2: list) -> list:
        return list(set(list1).intersection(set(list2)))

index_value = list(sentence_dict.keys())
search_key = ['this', 'first']
for key in search_key:
        index_value = list_intersection(index_dict[key], index_value)

print(index_value)
```

```
     [0]
```

```
import re

def create_index(docs):
        """
        Create an inverted index from a list of documents.
        """
        index = {}
        for i, doc in enumerate(docs):
                # Tokenize document and convert to lowercase
                tokens = re.findall(r'\b\w+\b', doc.lower())
                # Create dictionary of term frequencies
                tf = {}
                for token in tokens:
                        if token in tf:
                                tf[token] += 1
                        else:
                                tf[token] = 1
                # Add document to index
                for token in tf:
                        if token in index:
                                index[token].append((i, tf[token]))
                        else:
                                index[token] = [(i, tf[token])]
        return index

# Example usage
docs = [
        "This is the first document.",
        "This is the second document.",
        "And this is the third one.",
        "Is this the first document?",
]
index = create_index(docs)
print(index)
```

```
     {'this': [(0, 1), (1, 1), (2, 1), (3, 1)], 'is': [(0, 1), (1, 1), (2, 1), (3, 1)], 'the': [(0, 1), (1, 1), (2, 1), (3, 1)], 'first': [(0, 1), (3
```

```
import  requests
from  bs4  import  BeautifulSoup
import  json
i=0
link=[0]*10

for  genre_id  in  range(1,  2):

        url  =  f'https://movies.yahoo.com.tw/moviegenre_result.html?genre_id={genre_id}&page=1'
        #url  =  f"https://movies.yahoo.com.tw/moviegenre_result.html?genre_id=0&page={page}"
        r  =  requests.get(url)
        #  將網頁內容解析為  BeautifulSoup  物件
        soup  =  BeautifulSoup(r.text,  "html.parser")

        last_page  =  soup.select('ul  >  li')[-3].text

        print("最後一頁的頁碼為：",  last_page)
```

```
最後一頁的頁碼為：  142
```

```
import  requests
from  bs4  import  BeautifulSoup

url  =  "https://movies.yahoo.com.tw/moviegenre_result.html?genre_id=1&page=1"

#  下載網頁內容
r  =  requests.get(url)

#  將網頁內容解析為  BeautifulSoup  物件
soup  =  BeautifulSoup(r.text,  "html.parser")

last_page  =  soup.select('ul  >  li')[-3].text

print("最後一頁的頁碼為：",  last_page)
```

```
最後一頁的頁碼為：  142
```

```
last_page1  =  int(last_page)
type(last_page1)
```

```
int
```

```
type(int(last_page))
```

```
int
```

```
for  genre_id  in  range(1,  20):
        print(genre_id)
```

```
1
2
3
4
5
6
7
8
9
```