TMDB API MINI-PROJECT

Question-1:

TMDb API enables you to find out the latest information about TV Shows, Movies and the biggest names in entertainment sector for a marvellous and fun TV/Movie watching experience.

Find the 'id' of the movie "Andhadhun" using TMDb API.

```
[ ]  1 import requests
     2
     3 url='https://api.themoviedb.org/3/search/movie?api_key=43805ec494ed121077777c0f9b8b2d39&query=Andhadhun'
     4 data=requests.get(url)
     5
     6 data_json=data.json()['results'][0]['id']
     7 print(data_json)

534780
```

Code & Output for Question-1

Question-2:

Fetch the company id company 'Marvel Studios' using TMDb. Print the id.

```
[2]  1 import requests
     2
     3 url='https://api.themoviedb.org/3/search/company?api_key=43805ec494ed121077777c0f9b8b2d39&query=Marvel%20Studios'
     4 data=requests.get(url)
     5 data_json=data.json()['results'][0]['id']
     6
     7 print(data_json)

420
```

Code & Output for Question-2

Question-3:

Find the vote count and vote average of the movie "3 Idiots" using the TMDb API

```
[3]  1 import requests
     2 url='https://api.themoviedb.org/3/movie/20453?api_key=43805ec494ed121077777c0f9b8b2d39&language=en-US'
     3
     4 data=requests.get(url)
     5 data_json=data.json()
     6 vote_count=data_json['vote_count']
     7 vote_avg=data_json['vote_average']
     8
     9 print(vote_count,vote_avg)

    2054 7.985
```

Code & Output for Question-3

Question-4:

Fetch the names of top 5 similar movies to 'Inception' from the TMDb API.

```
 1 import requests
 2 api_key = "e226f4a5f5bace766952aa0d17182959"
 3 api_link = "https://api.themoviedb.org/3"
 4 params = {'query':"Inception", 'api_key':api_key}
 5 response = requests.get(api_link + "/search/movie", params=params)
 6 data = response.json()
 7 results = data.get('results')
 8 for result in results:
 9     if result.get('original_title') == 'Inception':
10         id = result.get('id')
11 # https://developers.themoviedb.org/3/movies/get-similar-movies
12
13 params2 = {'api_key':api_key}
14 response2 = requests.get(api_link + "/movie/" + str(id) + "/similar",
15 params=params2)
16 data2 = response2.json()
17 results2 = data2.get('results')
18 for result in results2[:5]:
19     print(result.get("title"))
```

```
Killing Zoe
Ed Wood
Terminator Salvation
Not Here to Be Loved
Transamerica
```

Code & Output for Question-4

Question-5:

Fetch the top-rated English movies in the US region using the TMDb API. From the result, print the first 10 movies which have original language as English. Also print their genres.

```python
import requests
api_key = "43805ec494ed121077777c0f9b8b2d39"
api_link = "https://api.themoviedb.org/3" #/movie/top_rated/apikey =
header = {'Accept': 'application/json'}
params = {'api_key':api_key, 'region':'US'}
response = requests.get(api_link + "/movie/top_rated", headers = header, params = params)


data = response.json()
results = data.get('results')
title_array = []
genre_id_array = []
for result in results:
    if result.get('original_language') == 'en':
        title_array.append(result.get('title'))
        genre_id_array.append(result.get('genre_ids'))

# To get the genre name corresponding to genre_id
response2 = requests.get(api_link + "/genre/movie/list", headers = header, params = params)
data2 = response2.json()
genres = data2.get('genres')
mapping = {}
for genre in genres:
    mapping[genre.get('id')] = genre.get('name')

for i in range(10):
    print(title_array[i], "-", end=" ")
    for id in genre_id_array[i]:
        print(mapping.get(id), end = ", ")
    print()
```

Code & Output for Question-5

```
The Godfather - Drama, Crime,
The Shawshank Redemption - Drama, Crime,
The Godfather Part II - Drama, Crime,
Schindler's List - Drama, History, War,
12 Angry Men - Drama,
The Dark Knight - Drama, Action, Crime, Thriller,
The Green Mile - Fantasy, Drama, Crime,
Pulp Fiction - Thriller, Crime,
The Boy, the Mole, the Fox and the Horse - Animation, Family, Adventure, Fantasy,
Forrest Gump - Comedy, Drama, Romance,
```

Code & Output for Question-5

Question-6:

Find the name and birthplace of the present most popular person according to TMDb API.

```
import requests
## Write your code here
api_key='43805ec494ed121077777c0f9b8b2d39'

url='https://api.themoviedb.org/3/person/popular?api_key=43805ec494ed121077777c0f9b8b2d39&language=en-US&page=1

data=requests.get(url)
data_json=data.json()['results'][0]

actor_id=data_json['id']
actor_name=data_json['name']


url_2='https://api.themoviedb.org/3/person/15737?api_key=43805ec494ed121077777c0f9b8b2d39&language=en-US'

data_2=requests.get(url_2)
bday_place=data_2.json()['place_of_birth']

print(actor_id)
print(actor_name,'-',bday_place)
```

```
15737
Helen McCrory - London, England, UK
```

Code & Output for Question-6

Question-7:

Fetch the overview of the TV Show "FRIENDS" using TMDb API.

```
import requests
api_key = "43805ec494ed121077777c0f9b8b2d39"
api_link = "https://api.themoviedb.org/3"
params = {'api_key':api_key,'query':'Friends'}
header = {'Accept': 'application/json'}
response2 = requests.get(api_link + "/search/tv", headers = header, params=params)
data=response2.json()
results=data.get('results')
for result in results:
    if result.get('name')=='Friends':
        print(result.get('overview'))
```

```
ix young people from New York City, on their own and struggling to survive in the real world
riends is a short-lived kids-oriented drama that aired in the spring of 1979. The series, wh
```

Code & Output for Question-7

Question-8:

Fetch the trending TV Shows for the week from the TMDb API and print the taglines of the top 5 shows. If there is no tagline, print 'Empty' instead

```
import requests
api_key = "43805ec494ed121077777c0f9b8b2d39"
api_link = "https://api.themoviedb.org/3"
params = {'api_key':api_key}
header = {'Accept': 'application/json'}
response = requests.get(api_link + "/trending/tv/week", headers = header, params = params)
data = response.json()
results = data.get("results")
ids=[]
for result in results[:5]:
    ids.append(result.get("id"))

for id in ids:
    response2 = requests.get(api_link + "/tv/" + str(id) , headers = header, params = params)
    data2 = response2.json()
    if (data2.get("tagline")) != "":
        print(data2.get("tagline"))
    else:
        print('Empty')
```

```
Empty
Bounty hunting is a complicated profession.
Revenge is best served raw.
When you're lost in the darkness, look for the light.
Empty
```

Code & Output for Question-8

Question-9:

Print the names of all the TV shows to be aired today whose original language is english.

```
import requests
## Write your code here
import requests as rq
page_num = 1
api_key = '43805ec494ed121077777c0f9b8b2d39'
api_link = 'https://api.themoviedb.org/3'
header = {'Accept':'application/json'}
params = {'language':'en','api_key':api_key}
r = rq.get(api_link+'/tv/airing_today',headers = header,params = params)
data = r.json()
# print(data)
res = data['results']
page_num = data.get('total_pages')
# print(page_num)
for i in range(1,page_num + 1):
    params = {'language':"en",'api_key':api_key,'page':i}
    r = rq.get(api_link+'/tv/airing_today',headers = header,params = params)
    data = r.json()
    results = data.get('results')
    for r in results:
        if r['original_language'] == 'en':
            print(r['name'])
```

Code & Output for Question-9