DATA COLLECTION INSTRUCTIONS

#install requests package if not already present in your system using 'pip install requests'
import requests
import time

1. **Creating headers for our request:**

```
headers = {
   'Referer': 'https://www.rottentomatoes.com/m/the_lion_king_2019/reviews?type=user',
   'User-Agent': 'Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like
Gecko) Chrome/74.0.3729.108 Safari/537.36',
   'X-Requested-With': 'XMLHttpRequest',
}
```

2. **URL:**
   It doesn't show any reviews data when accessed from browser by default. Use headers Referer if you want to view the data or its page source on browser.

```
url = 'https://www.rottentomatoes.com/napi/movie/9057c2cf-7cab-317f-876f-
e50b245ca76e/reviews/user'
```

3. **Initial payload parameters**

```
payload = {
   'direction': 'next',
   'endCursor': '',
   'startCursor': '',
}
```

4. **Creating a Session Object which is persistent to load multiple page reviews:**
   Note that it is initialization operation, hence will be done only once irrespective of how many pages you want to read from the server

```
s = requests.Session()
```

5. **The code to fetch one-page reviews (each page has 10 reviews) by using GET call on the required URL with our header and payload parameters**
   The response type is json so we can get the one page of reviews data in json format by calling its json function

```
r = s.get(url, headers=headers, params=payload)          # GET Call
data = r.json()
```

Sample format of 'data' is given as *sample_response_object.json*

You need to collect all the requested attributes from the 'data' object and populate a DataFrame with their values for each page. the fully populated DataFrame will be your train data.

To get the next page reviews, update the payload parameters 'startCursor' & 'endCursor' with their values from 'data' object and make a new GET call. Repeat the process till you collect 3000 reviews.

IMPORTANT NOTE:

You are suggested to run the following command (for 5sec lag – increase if required) when making each GET call in order to avoid your IP getting blocked by rotten tomatoes server. Note that INSOFE doesn't take responsibility for you being blocked by their server for making unusual requests and thereby failing to collect data. However, you may collect the data from another computer/IP when blocked.

time.sleep(5)