In [1]:

```
import requests as req
from bs4 import BeautifulSoup
from random import randint
from time import sleep
import random
from numpy import random
import pandas as pd
```

In [2]:

```
from selenium.webdriver import Chrome
from selenium.common.exceptions import TimeoutException
driver = Chrome(executable_path='chromedriver')
driver.set_page_load_timeout(10)
```

In [4]:

```
url = "https://www.metacritic.com/browse/movies/score/metascore/all/filtered?view=de
driver.get(url)
html = driver.page_source

s = BeautifulSoup(html, "lxml")
```

In [5]:

```
links = []
films = s.findAll('td', class_='clamp-summary-wrap')
for film in films:
    link = "https://www.metacritic.com"+film.find('a', class_='title').get('href')
    links.append([link+"/details"])
```

In [6]:

```
len(links)
```

Out[6]:

```
100
```

In [7]:

```
data = []

ban_count = 0;

for link in links:
    sleep(random.uniform(3, 7))

    try:
        driver.get(str(link)[2:-2])
    except TimeoutException:
        print("loading timeout")

    html = driver.page_source
    s = BeautifulSoup(html, "lxml")

    if ban_count>5: break

    try:
        title = s.find('div', class_='product_page_title oswald upper').find('h1').t
        date = s.find('table', class_='details').find('tr', class_='home_release_dat
        genre = s.find('table', class_='details').find('tr', class_='genres').find('
        metascore = s.find('span', class_='metascore_w larger movie positive perfect
        userscore = s.find('span', class_='metascore_w user larger movie positive').
        data.append([link, title, date, genre, metascore, userscore])

    except AttributeError:
        print('BAN!')
        ban_count = ban_count +1
```

```
BAN!
BAN!
BAN!
BAN!
BAN!
BAN!
```

In [8]:

```
data[0]
```

Out[8]:

```
[['https://www.metacritic.com/movie/citizen-kane/details'],
 'Citizen Kane',
 'Sep 25, 2001',
 'Drama',
 '100',
 '8.4']
```

In [9]:

```
head = ['link', 'title', 'date', 'genre', 'metascore', 'userscore']
dt = pd.DataFrame(data, columns=head)
dt.to_csv('metacritic_data.csv', sep=';', encoding='utf-16')
```

In [ ]:

```

```