

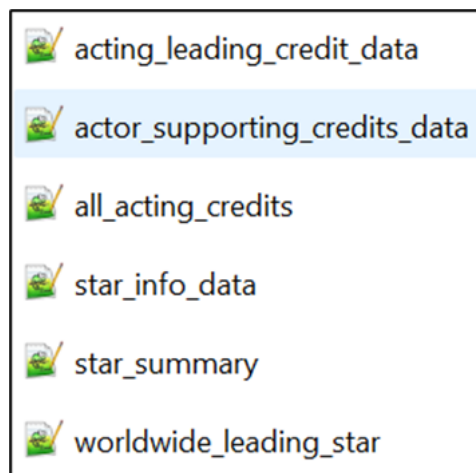


Movie Analysis Data Collection & EDA

▼ 1.Data Collection

แหล่งข้อมูลที่จะได้มาจาก 3 แหล่งข้อมูลหลักๆ

1. Top Stars in Leading Roles at the Worldwide Box Office จากเว็บไซต์ <https://www.the-numbers.com/box-office-star-records/worldwide/lifetime-acting/top-grossing-leading-stars> โดยใช้ Library Selenium ในการทำ Scrap ข้อมูลออกมา จำนวน 6 ไฟล์ซึ่งจะประกอบไปด้วยข้อมูลรายละเอียดของดารา Hollywood จำนวน 10,000 รายซึ่งจะประกอบไปด้วยข้อมูล หน้าที่ดาราคอนนั้นๆ เล่น วันเกิด บทบาทหลักๆ ในการ รายได้ทั้งหมดที่ดาราคอนนั้นได้สร้างไว้ในวงการ รายได้ทั้งหมดเฉลี่ยที่ดาราคอนนั้นได้สร้างไว้ในวงการ รายชื่อของดาราคอนที่ร่วมงานกันบ่อยๆ สิ่งสรุปภาพของดาราคอนนั้นๆ หนึ่งแต่ละเรื่องที่ดาราคอนนั้นเล่น วันที่หนังเข้าฉาย รายได้หนังแต่ละเรื่อง จำนวนโรงภาพยนตร์ที่เข้าฉาย



ตัวอย่างหน้าตาข้อมูลจากไฟล์ star_info_data.csv

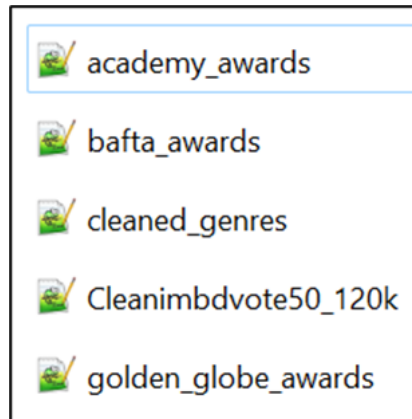
	Rank	Name	WorldwideBoxOffice	Movies	Average	Rank_u
0	1	Scarlett Johansson	\$14,520,720,009	33	\$440,021,818	1
1	2	Robert Downey, Jr.	\$14,393,065,910	44	\$327,115,134	2
2	3	Samuel L. Jackson	\$14,376,505,937	65	\$221,177,014	3
3	4	Zoe Saldana	\$14,207,267,921	31	\$458,298,965	4
4	5	Chris Pratt	\$13,872,208,523	23	\$603,139,501	5

หมายเหตุ : ข้อมูลไฟล์ต่างสามารถดูได้ในโฟลเดอร์

2.IMDB Database <https://www.imdb.com/>

ข้อมูลที่ได้จากการ Scrape เว็บ IMDB ซึ่งผ่านกระบวนการ Clean มาแล้วโดยใช้เครื่องมือสำเร็จรูปซึ่งประกอบไปด้วย

- ข้อมูลของหนัง และดาราที่ได้รับรางวัลของตั้งแต่ก่อตั้งจนถึงปี 2020 ของ 3 องค์การดังนี้ Academy Awards, Bafta Awards, Golden Globe Awards อีกสอง ไฟล์จะเป็นข้อมูลของหนังที่อยู่ใน IMDB
- ข้อมูลรายละเอียดของหนัง และข้อมูล Genre(ชนิดของหนัง) ของหนังแต่ละเรื่อง



หมายเหตุ : ข้อมูลไฟล์ต่างสามารถดูได้ในโฟลเดอร์

▼ 2.Cleansing, Transformation and EDA

การทำความสะอาดข้อมูลของเรานั้นจะกระทำกับข้อมูล Top Stars in Leading Roles at the Worldwide Box Office

โดยที่ขั้นตอนของการทำความสะอาดไฟล์แต่ละไฟล์จะมีความคล้ายกันอาจจะแตกต่างกันบ้างจึงขอยกตัวอย่างไฟล์ที่มีขั้นตอนการทำความสะอาดมากที่สุดคือไฟล์ actor_supporting_credits_data.csv ซึ่งประกอบไปด้วยขั้นตอนดังนี้

- Read Data - อ่านข้อมูลด้วย Pandas DataFrame

```
actor_supporting_credits_data = pd.read_csv(r'C:\Playground\DADS_5001_Tools\movie_pj\raw_data\actor_supporting_credits_data.csv',
display(actor_supporting_credits_data.head())
print(actor_supporting_credits_data.shape)
```

- View Data info - ดูรายละเอียดข้อมูลในแต่ละ Column

```
actor_supporting_credits_data.info()
actor_supporting_credits_data.describe(include = 'all')
```

- Check Duplicated Values and Deduplicate - ดูค่าซ้ำ และกำจัดค่าซ้ำ

```
display(actor_supporting_credits_data[ ['name','release_date','title','href_title'] ].value_counts().to_frame().query('count > 1 '))
display(actor_supporting_credits_data[actor_supporting_credits_data.duplicated()])
print(len('===== After Deduplicated =====')*'*')
print('===== After Deduplicated =====')
print(len('===== After Deduplicated =====')*'*')
actor_supporting_credits_data.drop_duplicates(subset = ['name','release_date','title','href_title'],keep = 'first,inplace = True)
display(actor_supporting_credits_data[ actor_supporting_credits_data['name'] == 'Robert Taylor'])
display(actor_supporting_credits_data[['name','release_date','title','href_title']].value_counts().to_frame().query('count > 1 '))
```

- Remove comma,\$ - ทำการแทนที่ ',' ด้วย คำว่าว่าง (") เพื่อที่จำสามารถเปลี่ยนชนิดของข้อมูลให้เป็นค่าตัวเลขได้เมื่อแทนที่เสร็จก็จะนำค่าใหม่ที่ได้ไปใส่ไว้ในคอลัมน์ใหม่ที่มี _u ต่อท้าย

```
cols_replace_comma = ['opening_weekend','max_theaters','domestic_box_office','worldwide_box_office']

## u = usable
```

```
# Remove , $
for i in cols_replace_comma:
    actor_supporting_credits_data[i+'_u'] = actor_supporting_credits_data[i].str.replace(',', ' ').str.replace('$', '')

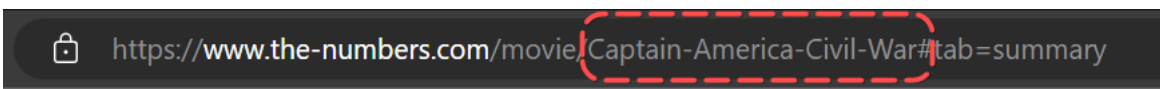
# Change data types for _u columns
for i in actor_supporting_credits_data.columns[actor_supporting_credits_data.columns.str.contains('_u')]:
    actor_supporting_credits_data[i] = pd.to_numeric( actor_supporting_credits_data[i] )

display( actor_supporting_credits_data.head() )
print(actor_supporting_credits_data.info())
```

- Extract movie_title from href_title column - เนื่องจากตัวเราต้องการชื่อนี้เต็มๆ ซึ่งไม่ถูกย่อที่มากับตัว Link ที่ต้องกดเข้าไปตรงชื่อนี้เพื่อให้ได้ชื่อนี้เต็มจากตัว Link จึงต้องใช้ตัว Regular Expression เข้ามาช่วยดึงเฉพาะชื่อนี้จากลิงค์ของหนังนั้นๆออกมาแล้วทำการแทนที่ “-” ด้วยช่องว่างแทน “ ” ก็จะได้ชื่อนี้ที่ต้องการออกมา

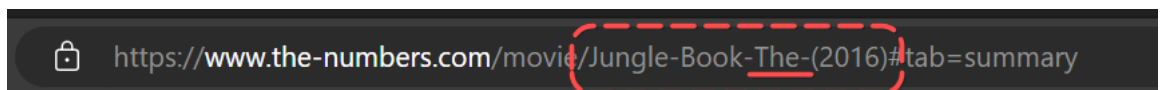
เมื่อคลิกไปที่ชื่อนี้จะนำไปสู่เว็บที่เป็นรายละเอียดของหนังเรื่องนั้นๆ

May 6, 2016	Captain America: Civil...	Natasha Romanoff ...	\$408,084,349	\$743,815,237	\$1,151,899,586
Apr 15, 2016	The Jungle Book	Kaa	\$364,001,123	\$589,534,698	\$953,535,821



```
actor_supporting_credits_data['movie_title'] = actor_supporting_credits_data['href_title'].str.extract(r'movie/(.*)#')
actor_supporting_credits_data['movie_title'] = actor_supporting_credits_data['movie_title'].str.replace('-', ' ')
```

Fix the movie_title column by removing the movies that end with 'The' and move 'The' to the front. - เนื่องจากปัญหาการที่เกิดจาก URL ของชื่อนี้ที่ขึ้นต้นด้วย “The” จะเกิดปัญหาคือใน URL ตัว The จะถูกนำไปอยู่ท้ายสุดของชื่อนี้ดังรูปต่อไปนี้



วิธีแก้ไขของเราคือใช้ Pandas เข้ามาช่วยด้วยการตรวจสอบว่าแถวนั้นๆ มี “The” อยู่ภายในหรือไม่ถ้ามีให้ลบออก แล้วนำมาเพิ่มส่วนหน้าสุด

```
actor_supporting_credits_data['movie_title'] = actor_supporting_credits_data.apply(lambda x : 'The ' + x['movie_title'].replace('The', ''))
```

- Convert domestic_share_str columns to float

domestic_share
NaN
55.8%
100.0%
29.1%
22.7%

จากคอลัมน์ในภาพเราจะทำการปรับให้ตัวเลขกลายเป็นรูปแบบทศนิยมด้วยการแทนที่ "%" ด้วย "" และแปลงเป็นค่า float64 หารด้วย 100

```
actor_supporting_credits_data['domestic_share_float'] = actor_supporting_credits_data['domestic_share'].str.rstrip('%').astype
```

domestic_share_float
NaN
0.558
1.000
0.291
0.227

- Strip all columns - ลบค่าว่างทั้งหมดหน้าหลังในคอลัมน์ที่เป็น datatype string ออกด้วย method strip

```
actor_supporting_credits_data = actor_supporting_credits_data.applymap(lambda x : x.strip() if isinstance(x,str) else x )
```

- Genrating cleaned file - ทำการเขียนไฟล์ที่ผ่านกระบวนการทั้งหมดออกมา

```
actor_supporting_credits_data.to_csv(f"C:\Playground\DADS_5001_Tools\movie_pj\cleaned_csv_file\c_actor_supporting_credits_data.csv")
display(pd.read_csv(f"C:\Playground\DADS_5001_Tools\movie_pj\cleaned_csv_file\c_actor_supporting_credits_data.csv", sep = '|').)
```