[MCU Worldwide Box Office Collection 데이터 분석 02]

- [생활-영화] Marvel Cinematic Universe 전세계 박스 오피스 컬렉션 데이터 세트
- 지역별 모든 박스 오피스 컬렉션 정보
- 데이터 출처 : https://www.kaggle.com/datasets/mayureshkoli/mcu-worldwide-box-office-collection (https://www.kaggle.com/datasets/mayureshkoli/mcu-worldwide-box-office-collection)
- 데이터 분석 코드
  - github 코드 (https://github.com/LDJWJ/dataAnalysis/blob/main/01_11_MCU_MOVIE_INFO.ipynb)
  - HTML코드 - 시작 (https://ldjwj.github.io/dataAnalysis/01_11_MCU_MOVIE_INFO.html)
  - HTML코드 - 전처리및탐색 (https://ldjwj.github.io/dataAnalysis/01_11_MCU_MOVIE_INFO_02.html)

학습 내용

- 관객수 시각화 - boxplot, histgram
- 시각화를 위한 기본 데이터 처리 - sum(), sort_values()

데이터 셋 개요

- 6개의 데이터 셋이 존재
- 데이터 파일
  - movie_info.csv : 영화 정보
  - asia_pacific_box_office.csv : 아시아 지역
  - europe_box_office.csv : 유럽 지역
  - middle_east_and_africa_box_office.csv : 중동, 아프리카 지역
  - north_america_box_office.csv : 북미 지역
  - south_america_box_office.csv : 남미 지역

데이터 설명

- Input/output variables

라이브러리 불러오기

In [1]:

```
import pandas as pd
import seaborn as sns
import numpy as np
import matplotlib.pyplot as plt

from sklearn.model_selection import train_test_split
```

데이터 불러오기

In [2]:

```
mov_info = pd.read_csv("./data/Marvel/movie_info.csv")
asia_info = pd.read_csv("./data/Marvel/asia_pacific_box_office.csv")
europe_info = pd.read_csv("./data/Marvel/europe_box_office.csv")
middle_east_info = pd.read_csv("./data/Marvel/middle_east_and_africa_box_office.csv")
north_america_info = pd.read_csv("./data/Marvel/north_america_box_office.csv")
south_america_info = pd.read_csv("./data/Marvel/south_america_box_office.csv")

mov_info.shape, asia_info.shape, europe_info.shape, middle_east_info.shape, north_america_info.shape
```

Out[2]:

```
((27, 11), (27, 17), (27, 31), (27, 13), (27, 5), (27, 12))
```

In [3]:

```
mov_info.head()
```

Out[3]:

	movie_title	release_date	season	phase	production_budget_in_million_(USD)	worldwide_col
0	Iron Man	May 2, 2008	Spring	1	140	
1	The Incredible Hulk	June 13, 2008	Spring	1	150	
2	Iron Man 2	May 7, 2010	Spring	1	200	
3	Thor	May 6, 2011	Spring	1	150	
4	Captain America: The First Avenger	July 22, 2011	Summer	1	140	

In [4]:

```
print( asia_info.head(3), end="₩n₩n" )
print( europe_info.head(3), end="₩n₩n" )
print( middle_east_info.head(3), end="₩n₩n" )
print( north_america_info.head(3), end="₩n₩n" )
print( south_america_info.head(3), end="₩n₩n" )
```

```
           movie_title  South Korea  Russia/CIS  Japan  Thailand  Indonesia  ₩
0             Iron Man        25.17        9.49   8.66      2.45       2.15   
1  The Incredible Hulk         6.38        6.41   1.69      1.18       1.50   
2           Iron Man 2        27.10       14.76  12.83      4.62       4.49   

   India  Taiwan  Philippines  Singapore  Vietnam  Malaysia  Hong Kong  ₩
0   1.99    5.37         3.99       3.82      NaN      3.47       2.84   
1   3.14    1.94         2.07       1.84     0.16      2.28       1.60   
2   1.23    4.04         6.25       4.19      NaN      4.64       3.76   

   New Zealand  Australia  China  Other_Asia_Pacific_Countries  
0         2.73      19.09  15.27                          1.37  
1         0.88       4.55   9.34                          0.70  
2         2.70      22.42   7.92                          6.57  

           movie_title  United Kingdom  Spain  Italy  Germany  Denmark  ₩
0             Iron Man           34.28  12.03  10.81     8.56     2.22   
1  The Incredible Hulk           15.16   7.69   6.46     2.46     1.10   
2           Iron Man 2           30.46   7.60   9.98     9.25     2.29   

   Hungary  Finland  Netherlands  Iceland  ...  Poland  Serbia and Montenegro  ₩
0     0.68     0.67         2.10     0.28  ...    1.00                   0.03   
1     0.31     0.22         1.34     0.15  ...    0.48                   0.05   
2     0.70     0.80         2.14     0.20  ...    1.14                   0.04   

   Estonia  Slovenia  Sweden  Belgium  Norway  Greece  France  ₩
0     0.08      0.09    2.06     1.97    1.86    1.80   19.20   
1     0.03      0.06    1.08     1.08    1.53    0.83    9.73   
2     0.08      0.12    1.97     1.89    2.46    1.57   19.79   

   Other_European_Countries  
0                      1.37  
1                      0.70  
2                      6.57  

[3 rows x 31 columns]

           movie_title  United Arab Emirates  Israel  South Africa  Nigeria  ₩
0             Iron Man                  1.84    0.61          1.46     0.05   
1  The Incredible Hulk                  1.81    0.49          0.93     0.03   
2           Iron Man 2                  2.25    0.68          2.59     0.06   

   Ghana  Kenya  East Africa  Lebanon  Egypt  Kuwait  Turkey  ₩
0    NaN    NaN         0.09     0.10   0.27    0.84    1.66   
1    NaN    NaN         0.08     0.10   0.28    0.66    1.02   
2   0.01    NaN         0.12     0.17   0.30     NaN    1.84   

   Other_Middle_East_and_African_Countries  
0                                     1.37  
1                                     0.70  
2                                     6.57  
```

```
           movie_title  USA_and_Canada  Mexico  Central America  Caribbean
0             Iron Man          319.03   15.95             1.37       1.37
1  The Incredible Hulk          134.81   12.65             0.70       0.70
2           Iron Man 2          312.43   18.40             6.57       6.57

           movie_title  Venezuela  Colombia  Bolivia  Uruguay  Peru  Paraguay  ₩
0             Iron Man       1.89      1.73     0.15     0.07  1.52       NaN
1  The Incredible Hulk       1.31      0.75     0.07     0.02  1.19       NaN
2           Iron Man 2       1.90      1.28     0.31     0.10  1.86       NaN

   Chile  Ecuador  Argentina  Brazil  Other_South_American_Countries
0   1.38     0.93       1.61   13.50                            1.37
1   0.34     0.38       0.94    4.89                            0.70
2   1.45     1.24       2.57   15.84                            6.57
```

In [5]:

```
asia_info.head()
```

Out[5]:

	movie_title	South Korea	Russia/CIS	Japan	Thailand	Indonesia	India	Taiwan	Philippines	Sing
0	Iron Man	25.17	9.49	8.66	2.45	2.15	1.99	5.37	3.99	
1	The Incredible Hulk	6.38	6.41	1.69	1.18	1.50	3.14	1.94	2.07	
2	Iron Man 2	27.10	14.76	12.83	4.62	4.49	1.23	4.04	6.25	
3	Thor	14.79	16.54	5.74	2.32	0.27	1.00	5.83	4.03	
4	Captain America: The First Avenger	3.81	8.64	3.43	2.48	2.05	0.12	6.32	3.58	

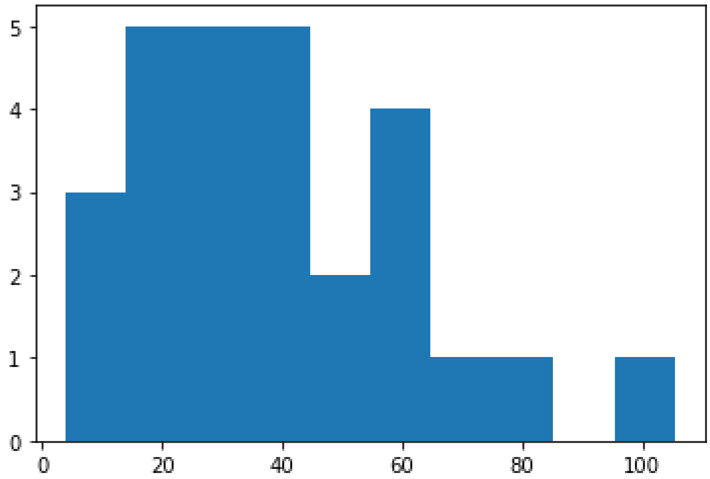
시각화

In [7]:

```
# 한국의 수치 히스토그램
plt.hist(asia_info['South Korea'])
```

Out[7]:

```
(array([3., 5., 5., 5., 2., 4., 1., 1., 0., 1.]),
 array([  3.81 ,  13.977,  24.144,  34.311,  44.478,  54.645,  64.812,
         74.979,  85.146,  95.313, 105.48 ]),
 <BarContainer object of 10 artists>)
```



In [9]:

```
asia_info.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 27 entries, 0 to 26
Data columns (total 17 columns):
 #   Column                        Non-Null Count  Dtype  
---  ------                        --------------  -----  
 0   movie_title                   27 non-null     object 
 1   South Korea                   27 non-null     float64
 2   Russia/CIS                    27 non-null     float64
 3   Japan                         27 non-null     float64
 4   Thailand                      27 non-null     float64
 5   Indonesia                     20 non-null     float64
 6   India                         26 non-null     float64
 7   Taiwan                        23 non-null     float64
 8   Philippines                   25 non-null     float64
 9   Singapore                     26 non-null     float64
 10  Vietnam                       13 non-null     float64
 11  Malaysia                      27 non-null     float64
 12  Hong Kong                     27 non-null     float64
 13  New Zealand                   27 non-null     float64
 14  Australia                     27 non-null     float64
 15  China                         21 non-null     float64
 16  Other_Asia_Pacific_Countries  27 non-null     float64
dtypes: float64(16), object(1)
memory usage: 3.7+ KB
```

아시아 국가의 관객수를 시각화 해보자.

관객수를 전부 더해서 마지막 행 더하기

In [35]:

```
dat = asia_info.sum()
dat
```

Out[35]:

```
movie_title                     Iron ManThe Incredible HulkIron Man 2ThorCapta...
South Korea                                                               1058.58
Russia/CIS                                                                 601.31
Japan                                                                      459.46
Thailand                                                                   183.87
Indonesia                                                                   225.3
India                                                                      303.99
Taiwan                                                                      256.8
Philippines                                                                227.72
Singapore                                                                  163.31
Vietnam                                                                     37.21
Malaysia                                                                   210.73
Hong Kong                                                                   248.6
New Zealand                                                                 90.75
Australia                                                                  683.18
China                                                                     3029.94
Other_Asia_Pacific_Countries                                               199.44
dtype: object
```

In [36]:

```
dat.index
```

Out[36]:

```
Index(['movie_title', 'South Korea', 'Russia/CIS', 'Japan', 'Thailand',
       'Indonesia', 'India', 'Taiwan', 'Philippines', 'Singapore', 'Vietnam',
       'Malaysia', 'Hong Kong', 'New Zealand', 'Australia', 'China',
       'Other_Asia_Pacific_Countries'],
      dtype='object')
```

In [37]:

```
### 1행부터 끝까지
dat = dat.iloc[1:]
dat
```

Out[37]:

```
South Korea                      1058.58
Russia/CIS                        601.31
Japan                             459.46
Thailand                          183.87
Indonesia                          225.3
India                             303.99
Taiwan                             256.8
Philippines                       227.72
Singapore                         163.31
Vietnam                            37.21
Malaysia                          210.73
Hong Kong                          248.6
New Zealand                        90.75
Australia                         683.18
China                            3029.94
Other_Asia_Pacific_Countries      199.44
dtype: object
```

In [38]:

```
dat.sort_values(ascending=False)
```

Out[38]:

```
China                            3029.94
South Korea                      1058.58
Australia                         683.18
Russia/CIS                        601.31
Japan                             459.46
India                             303.99
Taiwan                             256.8
Hong Kong                          248.6
Philippines                       227.72
Indonesia                          225.3
Malaysia                          210.73
Other_Asia_Pacific_Countries      199.44
Thailand                          183.87
Singapore                         163.31
New Zealand                        90.75
Vietnam                            37.21
dtype: object
```

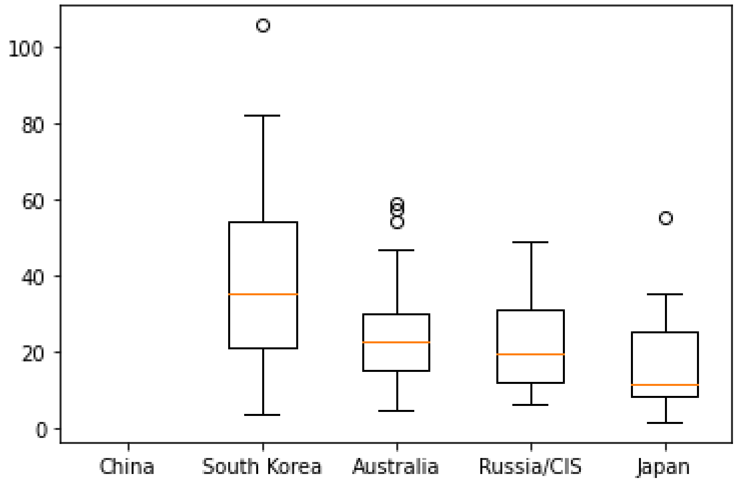
관객수 많은 5개국을 boxplot 확인

In [39]:

```
plt.boxplot( [asia_info['China'], asia_info['South Korea'],
              asia_info['Australia'], asia_info['Russia/CIS'],
              asia_info['Japan'] ],
             labels=['China', 'South Korea', 'Australia', 'Russia/CIS', 'Japan'])
```

Out[39]:

```
{'whiskers': [<matplotlib.lines.Line2D at 0x209dcc5dd90>,
  <matplotlib.lines.Line2D at 0x209dcc78130>,
  <matplotlib.lines.Line2D at 0x209dcc835b0>,
  <matplotlib.lines.Line2D at 0x209dcc83910>,
  <matplotlib.lines.Line2D at 0x209dcf9bdc0>,
  <matplotlib.lines.Line2D at 0x209dcfb6160>,
  <matplotlib.lines.Line2D at 0x209dcfcc5e0>,
  <matplotlib.lines.Line2D at 0x209dcfcc940>,
  <matplotlib.lines.Line2D at 0x209dd017df0>,
  <matplotlib.lines.Line2D at 0x209dd034190>],
 'caps': [<matplotlib.lines.Line2D at 0x209dcc78490>,
  <matplotlib.lines.Line2D at 0x209dcc787f0>,
  <matplotlib.lines.Line2D at 0x209dcc83c70>,
  <matplotlib.lines.Line2D at 0x209dcc83fd0>,
  <matplotlib.lines.Line2D at 0x209dcfb64c0>,
  <matplotlib.lines.Line2D at 0x209dcfb6820>,
  <matplotlib.lines.Line2D at 0x209dcfcccd0>,
  <matplotlib.lines.Line2D at 0x209dd017070>,
  <matplotlib.lines.Line2D at 0x209dd0344f0>,
  <matplotlib.lines.Line2D at 0x209dd034850>],
 'boxes': [<matplotlib.lines.Line2D at 0x209dcc5db50>,
  <matplotlib.lines.Line2D at 0x209dcc83250>,
  <matplotlib.lines.Line2D at 0x209dcf9ba30>,
  <matplotlib.lines.Line2D at 0x209dcfcc280>,
  <matplotlib.lines.Line2D at 0x209dd017a90>],
 'medians': [<matplotlib.lines.Line2D at 0x209dcc78b50>,
  <matplotlib.lines.Line2D at 0x209dcf9b370>,
  <matplotlib.lines.Line2D at 0x209dcfb6b80>,
  <matplotlib.lines.Line2D at 0x209dd0173d0>,
  <matplotlib.lines.Line2D at 0x209dd034bb0>],
 'fliers': [<matplotlib.lines.Line2D at 0x209dcc78eb0>,
  <matplotlib.lines.Line2D at 0x209dcf9b6d0>,
  <matplotlib.lines.Line2D at 0x209dcfb6ee0>,
  <matplotlib.lines.Line2D at 0x209dd017730>,
  <matplotlib.lines.Line2D at 0x209dd034f10>],
 'means': []}
```



- china는 결측치가 있어 표시가 되지 않음.

movie info를 이용한 선형회귀 모델 구축

In [7]:

```
mov_info.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 27 entries, 0 to 26
Data columns (total 11 columns):
 #   Column                                Non-Null Count  Dtype  
---  ------                                --------------  -----  
 0   movie_title                           27 non-null     object 
 1   release_date                          27 non-null     object 
 2   season                                27 non-null     object 
 3   phase                                 27 non-null     int64  
 4   production_budget_in_million_(USD)    27 non-null     int64  
 5   worldwide_collection_in_million_(USD) 27 non-null     float64
 6   tomatometer                           27 non-null     float64
 7   tomato_audience_score                 27 non-null     float64
 8   imdb                                  27 non-null     float64
 9   metascore                             27 non-null     float64
 10  meta_user_score                       27 non-null     float64
dtypes: float64(6), int64(2), object(3)
memory usage: 2.4+ KB
```

In [8]:

```
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
```

In [9]:

```
mov_info.head()
```

Out[9]:

	movie_title	release_date	season	phase	production_budget_in_million_(USD)	worldwide_col
0	Iron Man	May 2, 2008	Spring	1	140	
1	The Incredible Hulk	June 13, 2008	Spring	1	150	
2	Iron Man 2	May 7, 2010	Spring	1	200	
3	Thor	May 6, 2011	Spring	1	150	
4	Captain America: The First Avenger	July 22, 2011	Summer	1	140	

- meta_user_score 사용자 예측 모델

In [10]:

```
mov_info.columns
```

Out[10]:

```
Index(['movie_title', 'release_date', 'season', 'phase',
       'production_budget_in_million_(USD)',
       'worldwide_collection_in_million_(USD)', 'tomatometer',
       'tomato_audience_score', 'imdb', 'metascore', 'meta_user_score'],
      dtype='object')
```

In [11]:

```
sel = [ 'production_budget_in_million_(USD)',
       'worldwide_collection_in_million_(USD)', 'tomatometer',
       'tomato_audience_score', 'imdb', 'metascore'   ]

X = mov_info[sel]
y = mov_info['meta_user_score']

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.1, random_state=0)

X_train.shape, X_test.shape
```

Out[11]:

```
((24, 6), (3, 6))
```

In [16]:

```
model = LinearRegression()
model.fit(X_train, y_train)
pred = model.predict(X_test)

print( model.score(X_test, y_test) )
```

```
-0.7185104879973476
```

In [13]:

```
### MSE 구하기
np.mean(  (pred - y_test)**2 )
```

Out[13]:

```
0.5613800927457998
```

In [14]:

```
### MAE 구하기
np.mean(  np.abs(pred - y_test) )
```

Out[14]:

```
0.47688356031118584
```