# 아래의 코드셀을 완성하고 실행 결과를 포함하여 제출 하세요.

## **1. 데이터 다운로드 & 업로드 & 라이브러리 불러오기**

* [disney\_plus\_titles.csv 파일 다운로드 링크](https://www.kaggle.com/datasets/shivamb/disney-movies-and-tv-shows)
* 데이터 업로드

from google.colab import drive  
drive.mount('/content/drive')

Mounted at /content/drive

# [라이브러리 불러오기]  
# 1) numpy 로드하여 np로 사용  
# 2) pandas 로드하여 pd로 사용  
# 3) matplotlib.pyplot 로드하여 plt로 사용  
# 4) seaborn 로드하여 sns로 사용  
import numpy as np  
import pandas as pd  
import matplotlib.pyplot as plt  
import seaborn as sns

# [데이터 업로드]  
# 5) colab 업로드 메뉴를 통해 업로드 진행  
import csv  
# 6) disney.csv 다운로드 받아서 pandas의 함수 이용하여 파일 읽어서 disney변수에 저장  
disney = pd.read\_csv('/content/drive/MyDrive/Colab Notebooks/data/disney\_plus\_titles.csv')  
disney

{"summary":"{\n \"name\": \"disney\",\n \"rows\": 1450,\n \"fields\": [\n {\n \"column\": \"show\_id\",\n \"properties\": {\n \"dtype\": \"string\",\n \"num\_unique\_values\": 1450,\n \"samples\": [\n \"s539\",\n \"s755\",\n \"s50\"\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"type\",\n \"properties\": {\n \"dtype\": \"category\",\n \"num\_unique\_values\": 2,\n \"samples\": [\n \"TV Show\",\n \"Movie\"\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"title\",\n \"properties\": {\n \"dtype\": \"string\",\n \"num\_unique\_values\": 1450,\n \"samples\": [\n \"Homeward Bound II: Lost in San Francisco\",\n \"Boat Builders\"\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"director\",\n \"properties\": {\n \"dtype\": \"category\",\n \"num\_unique\_values\": 609,\n \"samples\": [\n \"Richard Fleischer\",\n \"Glenn Weiss\"\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"cast\",\n \"properties\": {\n \"dtype\": \"string\",\n \"num\_unique\_values\": 1193,\n \"samples\": [\n \"Thomas Nicholas, Joss Ackland, Art Malik, Paloma Baeza, Kate Winslet, Daniel Craig\",\n \"Hutch Dano, G. Hannelius, Maurice Godin, David Lambert, Kelsy Chow, Debra Mooney\"\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"country\",\n \"properties\": {\n \"dtype\": \"category\",\n \"num\_unique\_values\": 89,\n \"samples\": [\n \"United States, Ireland\",\n \"United States, Australia, France, Canada\"\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"date\_added\",\n \"properties\": {\n \"dtype\": \"object\",\n \"num\_unique\_values\": 167,\n \"samples\": [\n \"July 10, 2020\",\n \"May 29, 2020\"\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"release\_year\",\n \"properties\": {\n \"dtype\": \"number\",\n \"std\": 21,\n \"min\": 1928,\n \"max\": 2021,\n \"num\_unique\_values\": 90,\n \"samples\": [\n 1992,\n 1997\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"rating\",\n \"properties\": {\n \"dtype\": \"category\",\n \"num\_unique\_values\": 9,\n \"samples\": [\n \"TV-Y\",\n \"PG\"\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"duration\",\n \"properties\": {\n \"dtype\": \"category\",\n \"num\_unique\_values\": 158,\n \"samples\": [\n \"74 min\",\n \"32 Seasons\"\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"listed\_in\",\n \"properties\": {\n \"dtype\": \"category\",\n \"num\_unique\_values\": 329,\n \"samples\": [\n \"Comedy, Family, Musical\",\n \"Comedy, Variety\"\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"description\",\n \"properties\": {\n \"dtype\": \"string\",\n \"num\_unique\_values\": 1448,\n \"samples\": [\n \"A toy-stealing bully meets his match in a Lost & Found box.\",\n \"India From Above, a two-part aerial documentary of a nation and its people.\"\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n }\n ]\n}","variable\_name":"disney","type":"dataframe"}

## **2. 데이터 내용 확인**

* .columns : 컬럼명 확인
* .head(N) : 데이터의 상단 N개 행 출력
* .info() : 데이터에 대한 전반적인 정보 제공
  + 행과 열의 크기
  + 컬럼명
  + 컬럼별 결측치
  + 컬럼별 데이터 타입

# [데이터 내용 확인]  
# 1) 컬럼명을 확인하세요.  
disney.columns

Index(['show\_id', 'type', 'title', 'director', 'cast', 'country', 'date\_added',  
 'release\_year', 'rating', 'duration', 'listed\_in', 'description'],  
 dtype='object')

# [데이터 내용 확인]  
# 2) 상단의 행 3개를 출력하세요.  
disney.head(3)

{"summary":"{\n \"name\": \"disney\",\n \"rows\": 1450,\n \"fields\": [\n {\n \"column\": \"show\_id\",\n \"properties\": {\n \"dtype\": \"string\",\n \"num\_unique\_values\": 1450,\n \"samples\": [\n \"s539\",\n \"s755\",\n \"s50\"\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"type\",\n \"properties\": {\n \"dtype\": \"category\",\n \"num\_unique\_values\": 2,\n \"samples\": [\n \"TV Show\",\n \"Movie\"\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"title\",\n \"properties\": {\n \"dtype\": \"string\",\n \"num\_unique\_values\": 1450,\n \"samples\": [\n \"Homeward Bound II: Lost in San Francisco\",\n \"Boat Builders\"\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"director\",\n \"properties\": {\n \"dtype\": \"category\",\n \"num\_unique\_values\": 609,\n \"samples\": [\n \"Richard Fleischer\",\n \"Glenn Weiss\"\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"cast\",\n \"properties\": {\n \"dtype\": \"string\",\n \"num\_unique\_values\": 1193,\n \"samples\": [\n \"Thomas Nicholas, Joss Ackland, Art Malik, Paloma Baeza, Kate Winslet, Daniel Craig\",\n \"Hutch Dano, G. Hannelius, Maurice Godin, David Lambert, Kelsy Chow, Debra Mooney\"\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"country\",\n \"properties\": {\n \"dtype\": \"category\",\n \"num\_unique\_values\": 89,\n \"samples\": [\n \"United States, Ireland\",\n \"United States, Australia, France, Canada\"\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"date\_added\",\n \"properties\": {\n \"dtype\": \"object\",\n \"num\_unique\_values\": 167,\n \"samples\": [\n \"July 10, 2020\",\n \"May 29, 2020\"\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"release\_year\",\n \"properties\": {\n \"dtype\": \"number\",\n \"std\": 21,\n \"min\": 1928,\n \"max\": 2021,\n \"num\_unique\_values\": 90,\n \"samples\": [\n 1992,\n 1997\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"rating\",\n \"properties\": {\n \"dtype\": \"category\",\n \"num\_unique\_values\": 9,\n \"samples\": [\n \"TV-Y\",\n \"PG\"\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"duration\",\n \"properties\": {\n \"dtype\": \"category\",\n \"num\_unique\_values\": 158,\n \"samples\": [\n \"74 min\",\n \"32 Seasons\"\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"listed\_in\",\n \"properties\": {\n \"dtype\": \"category\",\n \"num\_unique\_values\": 329,\n \"samples\": [\n \"Comedy, Family, Musical\",\n \"Comedy, Variety\"\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"description\",\n \"properties\": {\n \"dtype\": \"string\",\n \"num\_unique\_values\": 1448,\n \"samples\": [\n \"A toy-stealing bully meets his match in a Lost & Found box.\",\n \"India From Above, a two-part aerial documentary of a nation and its people.\"\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n }\n ]\n}","variable\_name":"disney","type":"dataframe"}

# 3) disney 데이터에 대한 전반적인 정보를 제공하는 함수를 사용하여 내용을 파악하세요.  
disney.info()

<class 'pandas.core.frame.DataFrame'>  
RangeIndex: 1450 entries, 0 to 1449  
Data columns (total 12 columns):  
 # Column Non-Null Count Dtype   
--- ------ -------------- -----   
 0 show\_id 1450 non-null object  
 1 type 1450 non-null object  
 2 title 1450 non-null object  
 3 director 977 non-null object  
 4 cast 1260 non-null object  
 5 country 1231 non-null object  
 6 date\_added 1447 non-null object  
 7 release\_year 1450 non-null int64   
 8 rating 1447 non-null object  
 9 duration 1450 non-null object  
 10 listed\_in 1450 non-null object  
 11 description 1450 non-null object  
dtypes: int64(1), object(11)  
memory usage: 136.1+ KB

## **3. 결측치 비율 확인**

# 1) for 반복문과 isna() 함수를 이용하여 결측치 비율을 출력하세요.  
  
for i in disney.columns :  
 missing\_rate = disney[i].isna().sum() / len(disney) \* 100  
 if missing\_rate > 0 :  
 print ("{} - missing\_rate : {}%".format(i, round(missing\_rate, 2)))

director - missing\_rate : 32.62%  
cast - missing\_rate : 13.1%  
country - missing\_rate : 15.1%  
date\_added - missing\_rate : 0.21%  
rating - missing\_rate : 0.21%

## **4. 결측치 처리**

# 1) director 변수에서 결측치를 fillna 이용하여 'No Data'로 처리하세요.  
# 결측치 처리 (1) - fillna  
# .fillna( ) : 결측치를 특정 값으로 채우거나 대체하여 처리  
# 원본 객체를 변경하려면 inplace = True 옵션 추가  
disney['director'].fillna('No Data', inplace = True)  
disney

# 2) cast, country 변수에서 결측치를 replace 이용하여 'No Data'로 처리해주세요.  
disney['cast'].replace(np.nan, 'No Data', inplace = True)  
disney['country'].replace(np.nan, 'No Data', inplace = True)  
disney

{"summary":"{\n \"name\": \"disney\",\n \"rows\": 1444,\n \"fields\": [\n {\n \"column\": \"show\_id\",\n \"properties\": {\n \"dtype\": \"string\",\n \"num\_unique\_values\": 1444,\n \"samples\": [\n \"s417\",\n \"s320\",\n \"s1038\"\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"type\",\n \"properties\": {\n \"dtype\": \"category\",\n \"num\_unique\_values\": 2,\n \"samples\": [\n \"TV Show\",\n \"Movie\"\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"title\",\n \"properties\": {\n \"dtype\": \"string\",\n \"num\_unique\_values\": 1444,\n \"samples\": [\n \"D2: The Mighty Ducks\",\n \"Toy Story: That Time Forgot\"\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"director\",\n \"properties\": {\n \"dtype\": \"category\",\n \"num\_unique\_values\": 610,\n \"samples\": [\n \"Jacob Cobi Benattia, Pete Berg, James Buchanan\",\n \"Steven Hunter\"\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"cast\",\n \"properties\": {\n \"dtype\": \"string\",\n \"num\_unique\_values\": 1189,\n \"samples\": [\n \"Maite Jauregui\",\n \"Steve Blum, Chris Edgerly, Jennifer Hale, Michael Ironside, Danielle Judovits, AJ Buckley\"\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"country\",\n \"properties\": {\n \"dtype\": \"category\",\n \"num\_unique\_values\": 90,\n \"samples\": [\n \"United Kingdom, France\",\n \"United Kingdom\"\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"date\_added\",\n \"properties\": {\n \"dtype\": \"object\",\n \"num\_unique\_values\": 166,\n \"samples\": [\n \"July 10, 2020\",\n \"February 14, 2020\"\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"release\_year\",\n \"properties\": {\n \"dtype\": \"number\",\n \"std\": 21,\n \"min\": 1928,\n \"max\": 2021,\n \"num\_unique\_values\": 90,\n \"samples\": [\n 1992,\n 1997\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"rating\",\n \"properties\": {\n \"dtype\": \"category\",\n \"num\_unique\_values\": 9,\n \"samples\": [\n \"TV-Y\",\n \"PG\"\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"duration\",\n \"properties\": {\n \"dtype\": \"category\",\n \"num\_unique\_values\": 158,\n \"samples\": [\n \"74 min\",\n \"32 Seasons\"\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"listed\_in\",\n \"properties\": {\n \"dtype\": \"category\",\n \"num\_unique\_values\": 328,\n \"samples\": [\n \"Animation, Comedy, Dance\",\n \"Animation, Superhero\"\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"description\",\n \"properties\": {\n \"dtype\": \"string\",\n \"num\_unique\_values\": 1442,\n \"samples\": [\n \"Violetta is a talented teen following her dream.\",\n \"Rock out to music videos from The Ghoul Girls!\"\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n }\n ]\n}","variable\_name":"disney","type":"dataframe"}

# 3) 결측치가 존재할 경우 dropna 이용하여 행 전체를 제거해주세요.  
disney.dropna(axis = 0, inplace = True)  
disney

{"summary":"{\n \"name\": \"disney\",\n \"rows\": 1444,\n \"fields\": [\n {\n \"column\": \"show\_id\",\n \"properties\": {\n \"dtype\": \"string\",\n \"num\_unique\_values\": 1444,\n \"samples\": [\n \"s417\",\n \"s320\",\n \"s1038\"\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"type\",\n \"properties\": {\n \"dtype\": \"category\",\n \"num\_unique\_values\": 2,\n \"samples\": [\n \"TV Show\",\n \"Movie\"\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"title\",\n \"properties\": {\n \"dtype\": \"string\",\n \"num\_unique\_values\": 1444,\n \"samples\": [\n \"D2: The Mighty Ducks\",\n \"Toy Story: That Time Forgot\"\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"director\",\n \"properties\": {\n \"dtype\": \"category\",\n \"num\_unique\_values\": 610,\n \"samples\": [\n \"Jacob Cobi Benattia, Pete Berg, James Buchanan\",\n \"Steven Hunter\"\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"cast\",\n \"properties\": {\n \"dtype\": \"string\",\n \"num\_unique\_values\": 1189,\n \"samples\": [\n \"Maite Jauregui\",\n \"Steve Blum, Chris Edgerly, Jennifer Hale, Michael Ironside, Danielle Judovits, AJ Buckley\"\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"country\",\n \"properties\": {\n \"dtype\": \"category\",\n \"num\_unique\_values\": 90,\n \"samples\": [\n \"United Kingdom, France\",\n \"United Kingdom\"\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"date\_added\",\n \"properties\": {\n \"dtype\": \"object\",\n \"num\_unique\_values\": 166,\n \"samples\": [\n \"July 10, 2020\",\n \"February 14, 2020\"\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"release\_year\",\n \"properties\": {\n \"dtype\": \"number\",\n \"std\": 21,\n \"min\": 1928,\n \"max\": 2021,\n \"num\_unique\_values\": 90,\n \"samples\": [\n 1992,\n 1997\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"rating\",\n \"properties\": {\n \"dtype\": \"category\",\n \"num\_unique\_values\": 9,\n \"samples\": [\n \"TV-Y\",\n \"PG\"\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"duration\",\n \"properties\": {\n \"dtype\": \"category\",\n \"num\_unique\_values\": 158,\n \"samples\": [\n \"74 min\",\n \"32 Seasons\"\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"listed\_in\",\n \"properties\": {\n \"dtype\": \"category\",\n \"num\_unique\_values\": 328,\n \"samples\": [\n \"Animation, Comedy, Dance\",\n \"Animation, Superhero\"\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"description\",\n \"properties\": {\n \"dtype\": \"string\",\n \"num\_unique\_values\": 1442,\n \"samples\": [\n \"Violetta is a talented teen following her dream.\",\n \"Rock out to music videos from The Ghoul Girls!\"\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n }\n ]\n}","variable\_name":"disney","type":"dataframe"}

## **5. 결측치 개수 확인**

disney.info()

<class 'pandas.core.frame.DataFrame'>  
Int64Index: 1444 entries, 0 to 1449  
Data columns (total 12 columns):  
 # Column Non-Null Count Dtype   
--- ------ -------------- -----   
 0 show\_id 1444 non-null object  
 1 type 1444 non-null object  
 2 title 1444 non-null object  
 3 director 1444 non-null object  
 4 cast 1444 non-null object  
 5 country 1444 non-null object  
 6 date\_added 1444 non-null object  
 7 release\_year 1444 non-null int64   
 8 rating 1444 non-null object  
 9 duration 1444 non-null object  
 10 listed\_in 1444 non-null object  
 11 description 1444 non-null object  
dtypes: int64(1), object(11)  
memory usage: 178.9+ KB

# 결측치 개수를 출력하여 결측치가 잘 처리되었는지 확인하세요.  
# isna().sum()이나 isnull().sum()이용  
disney.isnull().sum()

show\_id 0  
type 0  
title 0  
director 0  
cast 0  
country 0  
date\_added 0  
release\_year 0  
rating 0  
duration 0  
listed\_in 0  
description 0  
dtype: int64

## **6. Feature Engineering**

* Feature Engineering : 기존에 존재하는 변수를 활용하여 새로운 정보를 추가로 생성하는 과정
* date\_added 변수를 이용하여 year\_added(개봉한 연도)와 month\_added(개봉한 월) 정보를 변수로 생성
  + .to\_datetime() : 시간 형식의 object 타입의 컬럼을 datetime 타입으로 변환
  + .dt.year : datetime에서 연도 정보 추출
  + .dt.month : datetime에서 월 정보 추출

# 1) date\_added 변수를 시간 형식의 object 타입의 컬럼에서 datetime 타입으로 변환하여 date\_added 변수를 출력하세요.  
disney['date\_added'] = pd.to\_datetime(disney['date\_added'])  
disney.info()

# 2) 타입이 변환된 disney의 date\_added 변수를 이용하여 year\_added(개봉한 연도) 변수 생성하세요.  
disney['year\_added'] = disney['date\_added'].dt.year  
  
# 3) 타입이 변환된 disney의 date\_added 변수를 이용하여 month\_added(개봉한 월) 변수 생성하세요.  
disney['month\_added'] = disney['date\_added'].dt.month  
  
  
# 4) 상단의 행 3개를 출력하세요.  
disney.head(3)

{"summary":"{\n \"name\": \"disney\",\n \"rows\": 1444,\n \"fields\": [\n {\n \"column\": \"show\_id\",\n \"properties\": {\n \"dtype\": \"string\",\n \"num\_unique\_values\": 1444,\n \"samples\": [\n \"s417\",\n \"s320\",\n \"s1038\"\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"type\",\n \"properties\": {\n \"dtype\": \"category\",\n \"num\_unique\_values\": 2,\n \"samples\": [\n \"TV Show\",\n \"Movie\"\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"title\",\n \"properties\": {\n \"dtype\": \"string\",\n \"num\_unique\_values\": 1444,\n \"samples\": [\n \"D2: The Mighty Ducks\",\n \"Toy Story: That Time Forgot\"\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"director\",\n \"properties\": {\n \"dtype\": \"category\",\n \"num\_unique\_values\": 610,\n \"samples\": [\n \"Jacob Cobi Benattia, Pete Berg, James Buchanan\",\n \"Steven Hunter\"\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"cast\",\n \"properties\": {\n \"dtype\": \"string\",\n \"num\_unique\_values\": 1189,\n \"samples\": [\n \"Maite Jauregui\",\n \"Steve Blum, Chris Edgerly, Jennifer Hale, Michael Ironside, Danielle Judovits, AJ Buckley\"\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"country\",\n \"properties\": {\n \"dtype\": \"category\",\n \"num\_unique\_values\": 90,\n \"samples\": [\n \"United Kingdom, France\",\n \"United Kingdom\"\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"date\_added\",\n \"properties\": {\n \"dtype\": \"date\",\n \"min\": \"2019-10-01 00:00:00\",\n \"max\": \"2021-11-26 00:00:00\",\n \"num\_unique\_values\": 166,\n \"samples\": [\n \"2020-07-10 00:00:00\",\n \"2020-02-14 00:00:00\"\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"release\_year\",\n \"properties\": {\n \"dtype\": \"number\",\n \"std\": 21,\n \"min\": 1928,\n \"max\": 2021,\n \"num\_unique\_values\": 90,\n \"samples\": [\n 1992,\n 1997\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"rating\",\n \"properties\": {\n \"dtype\": \"category\",\n \"num\_unique\_values\": 9,\n \"samples\": [\n \"TV-Y\",\n \"PG\"\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"duration\",\n \"properties\": {\n \"dtype\": \"category\",\n \"num\_unique\_values\": 158,\n \"samples\": [\n \"74 min\",\n \"32 Seasons\"\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"listed\_in\",\n \"properties\": {\n \"dtype\": \"category\",\n \"num\_unique\_values\": 328,\n \"samples\": [\n \"Animation, Comedy, Dance\",\n \"Animation, Superhero\"\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"description\",\n \"properties\": {\n \"dtype\": \"string\",\n \"num\_unique\_values\": 1442,\n \"samples\": [\n \"Violetta is a talented teen following her dream.\",\n \"Rock out to music videos from The Ghoul Girls!\"\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"year\_added\",\n \"properties\": {\n \"dtype\": \"number\",\n \"std\": 0,\n \"min\": 2019,\n \"max\": 2021,\n \"num\_unique\_values\": 3,\n \"samples\": [\n 2021,\n 2020\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"month\_added\",\n \"properties\": {\n \"dtype\": \"number\",\n \"std\": 3,\n \"min\": 1,\n \"max\": 12,\n \"num\_unique\_values\": 12,\n \"samples\": [\n 1,\n 2\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n }\n ]\n}","variable\_name":"disney","type":"dataframe"}

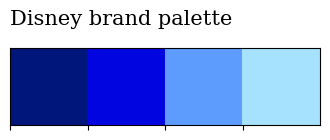
# rating 변수의 값의 종류 파악

# 시청 등급을 참고하여 age\_group 변수 생성  
  
# disney['age\_group']를 생성하여 disney['rating'] 값 삽입  
disney['age\_group'] = disney['rating']  
  
# age\_group 변수에 딕셔너리로 시청 등급에 대한 key, value 선언  
age\_group = {'TV-G': 'Kids',  
 'PG': 'Older Kids',  
 'TV-PG': 'Older Kids',  
 'PG-13': 'Teens',  
 'TV-14': 'Young Adults',  
 'G': 'Kids',  
 'TV-Y7': 'Older Kids',  
 'TV-Y': 'Kids',  
 'TV-Y7-FV': 'Older Kids'  
 }  
  
# .map( ) : 사전에 정의한 내용을 변수에 적용하기  
# .head()로 확인하기

## **7. type: Movie & TV show 시각화**

* plt.pie()
* sns.countplot()
* plt.fill\_between()

# 디즈니 색상 팔렛트  
sns.palplot(['#00167b','#0106e1','#5d9cfc', '#a6e2fe'])  
plt.title("Disney brand palette ", loc='left', fontfamily='serif', fontsize=15, y=1.2)  
plt.show()



# Movies & TV shows의 각각 value\_counts 출력  
# .value\_counts( ) : 값의 개수 출력

# 위 문장의 개수(vjalue\_counts()결과)를 전체변환하여 ratio 변수에 저장  
# .T : 전치(Transpose) 변환

# pie plot 통해 Movies & TV shows 각각 비율을 파이차트로 시각화  
  
# plt.pie의 매개변수  
# labels : 부채꼴 조각 이름  
# autopct : 부채꼴 안에 표시될 숫자 형식 지정  
# 문자열에서 % 포맷팅으로 %0.f 형태로 사용하면 소수점 없이 정수처럼 인식  
# 진짜 %를 표시하기 위해 %%로 작성  
# startangle : 부채꼴이 그려지는 시작 각도 설정, 90이면 12시 방향  
# explode : 부채꼴이 파이 플롯의 중심에서 벗어나는 정도 설정  
# shadow : 그림자 효과 표시  
  
  
  
  
# 제목 표시하세요.  
# plt.suptitle : 전체 제목 : Movie & TV Show distribution  
# plt.title : 서브 제목 : More movies than TV shows on Disney

# 연도별 Movies & TV shows 수치 시각화  
  
# countplot( ) : 각 범주에 속하는 데이터의 개수를 막대 그래프 시각화  
# data : countplot에서 사용할 데이터 셋  
# x : x축 설정  
# hue : 특정 열 데이터로 색상을 구분하여 출력

# 월별 Movie & TV show 수치 시각화  
# .groupby( ) : 그룹별 집계, 개봉한 월로 묶고 타입의 개수 구하기  
# .value\_counts( ) : 값의 개수 출력  
# .unstack( ) : 인덱스를 컬럼으로 바꾸는 역할

# 월별로 Movies & TV shows 수치 시각화  
# plt.fill\_between() 이용  
  
# fill\_between() : x축을 기준으로 그래프 영역을 채우는 함수  
# x : 곡선을 정의하는 노드의 x 좌표  
# y1 : 첫 번째 곡선을 정의하는 노드의 y 좌표  
# y2 : 두 번째 곡선을 정의하는 노드의 y 좌표  
# alpha : 투명도  
# label : 'Movie', 'TV Show' 문자열 입력  
# legend() 함수를 사용해서 그래프에 범례 표시  
  
# xticks(x, month\_name) : x축의 눈금 레이블에 month\_name 값의 순서대로 설정

# 월별 Movie & TV show 수치 시각화  
# 1) plt.figure 함수를 이용하여 figsize를 (15,5) 설정  
# 2) sns.countplot 함수를 이용하여 월별 Movies & TV shows 시각화 완성하기  
# 3) suptitle에는 'Movie & TV Show added by month' 입력하여 그래프 전체 제목 출력하기  
# 4) title에는 'The most Movies and TV shows were added in November.' 입력하여 출력하기

## **8. DisneyPlus 나라별 타겟팅 연령 시각화**

* heatmap( ) : 직사각형 데이터를 색으로 행렬을 표현하여 시각화
* cmap : 시각화 시키는 컬러맵
* linewidth : 각 셀을 분할할 선의 너비
* annot : True라면 각 셀에 데이터 값 표시
* fmt : 문자열 형식화 코드 .0% = N .1% = N.n .2% = N.nn
* plt.matplotlib.colors.LinearSegmentedColormap.from\_list( ) : 컬러맵 생성

# 나라별 타겟팅하는 연령 수치화해서 disney\_age\_country변수에 할당  
# age\_group 으로 묶고 나라(country)의 개수 구하기  
# .groupby( ) : 그룹화  
# .value\_counts( ) : 값의 개수 출력  
# .unstack( ) : 인덱스를 컬럼으로 바꾸는 역할

# 특정 나라별 타겟팅하는 특정 연령 선택하여 수치화  
  
# 특정 나이대 : ['Kids','Older Kids','Teens','Young Adults']  
# 특정 나라 : ['United States', 'India', 'United Kingdom', 'Canada', 'France', 'Germany']  
age\_order = ['Kids','Older Kids','Teens','Young Adults']  
country\_order = ['United States', 'India', 'United Kingdom', 'Canada',  
 'France', 'Germany']  
  
# .loc[] : 행 이름age\_order과 열 이름country\_order을 사용하여 특정 행과 열 선택  
# .fillna(0, inplace = True) : 결측치를 0으로 대체하여 원본 데이터에 저장 (inplace = True)

# 나라별 타겟팅하는 연령별 비율(disney\_age\_country)을 알기 위해  
# 열(나라)마다 총합을 구하여 각각 행을 나누어서 연령별 비율 구하기  
# .sum(axis=0) : 열 기준으로 더하기  
# .div(axis=1) : 행 기준으로 나누기

# 나라별 타겟팅하는 연령을 히트맵 그리기  
  
# cmap : 시각화 시키는 컬러맵  
# linewidth : 각 셀을 분할할 선의 너비  
# annot : True라면 각 셀에 데이터 값 표시  
# fmt : 문자열 형식화 코드  
# .0% = X  
# .1% = X.x  
# .2% = X.xx  
# plt.matplotlib.colors.LinearSegmentedColormap.from\_list( ) : 컬러맵 생성

## **9. 워드 클라우드**

* title 변수를 이용한 워드 클라우드 만들기
* disney.jpg를 다운받아, 코랩에 업로드하기

# 워드 클라우드 생성에 필요한 모듈 선언  
  
from wordcloud import WordCloud  
from PIL import Image  
import PIL  
import matplotlib.pyplot as plt

# Disney 데이터의 title의 단어를 이용한 워드 클라우드 생성  
  
# 1) wordcolud에서 작동할 수 있도록 데이터프레임을 list로 1차 변환시키고 str(문자열)로 2차 변환하여 text 변수에 저장  
# 2) plt.figure 함수를 이용하여 figsize를 (10,10) 설정  
# 3) disney.jpg 열어서 단어를 그릴 위치 설정하여 mask\_img 변수에 저장(흰색(#FFFFFF) 항목은 마스킹된 것으로 간주)  
# 4) plt.matplotlib.colors.LinearSegmentedColormap.from\_list("", [컬러코드리스트])로 컬러맵 생성하여 cmap 변수에 저장  
# 5) WordCloud( ).generate(text) : 선언해준 text에서 wordcloud를 생성  
# (background\_color = 'white', width = 1000, height = 1000, max\_words = 300)  
# 6) title을 'Keywords in the title of Movies and TV shows' 로 출력 (fontweight='bold', fontfamily='serif', fontsize=20) 적용  
# 7) plt.imshow( ) : 이미지로 표시  
# 8) plt.axis('off') : 축 삭제  
# 9) plt.show()

# 노트북 파일과 워드파일로 저장하여 두가지 파일 모두 제출하세요.  
  
# 아래의 문장을 실행시켜 워드파일로 저장한 후, 다운로드하여 제출.  
# !pandoc 파일경로/파일이름.ipynb -s -o 파일경로/disneyreport\_본인이름.docx