https://colab.research.google.com/drive/1H07GB4MAtQPFP4DL6_U_HJtvHrzFheuh?usp=sharing

```
# IMDB BC Intro
```

```
import pandas as pd
import numpy as np
```

```
!gdown 1s2TkjSpzNc4SyxqRrQleZyDIHlc7bxnd
```

```
Downloading...
From: https://drive.google.com/uc?id=1s2TkjSpzNc4SyxqRrQleZyDIHlc7bxnd
To: /content/movies.csv
100% 112k/112k [00:00<00:00, 65.9MB/s]
```

```
!gdown 1Ws-_s1fHZ9nHfGLVUQurbHDvStePlEJm
```

```
Downloading...
From: https://drive.google.com/uc?id=1Ws-_s1fHZ9nHfGLVUQurbHDvStePlEJm
To: /content/directors.csv
100% 65.4k/65.4k [00:00<00:00, 55.3MB/s]
```

```
movies = pd.read_csv("movies.csv")  #index_col=0
movies.head()
```

	Unnamed: 0	id	budget	popularity	revenue	title	vote_average	vote_count	director_i
0	0	43597	237000000	150	2787965087	Avatar	7.2	11800	476
1	1	43598	300000000	139	961000000	Pirates of the Caribbean: At World's End	6.9	4500	476
2	2	43599	245000000	107	880674609	Spectre	6.3	4466	476
3	3	43600	250000000	112	1084939099	The Dark Knight Rises	7.6	9106	476
4	5	43602	258000000	115	890871626	Spider-Man 3	5.9	3576	476

```
movies.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1465 entries, 0 to 1464
Data columns (total 12 columns):
 #   Column        Non-Null Count  Dtype
---  ------        --------------  -----
 0   Unnamed: 0    1465 non-null   int64
 1   id            1465 non-null   int64
 2   budget        1465 non-null   int64
 3   popularity    1465 non-null   int64
 4   revenue       1465 non-null   int64
 5   title         1465 non-null   object
 6   vote_average  1465 non-null   float64
 7   vote_count    1465 non-null   int64
 8   director_id   1465 non-null   int64
 9   year          1465 non-null   int64
 10  month         1465 non-null   object
 11  day           1465 non-null   object
dtypes: float64(1), int64(8), object(3)
memory usage: 137.5+ KB
```

```
movies.drop("Unnamed: 0", axis=1, inplace=True)
```

```
movies.head()
```

	id	budget	popularity	revenue	title	vote_average	vote_count	director_id	year	mon
0	43597	237000000	150	2787965087	Avatar	7.2	11800	4762	2009	D
1	43598	300000000	139	961000000	Pirates of the Caribbean: At World's End	6.9	4500	4763	2007	M
2	43599	245000000	107	880674609	Spectre	6.3	4466	4764	2015	C
3	43600	250000000	112	1084939099	The Dark Knight Rises	7.6	9106	4765	2012	
4	43602	258000000	115	890871626	Spider-Man 3	5.9	3576	4767	2007	M

```
movies.shape
```

```
(1465, 11)
```

```
directors = pd.read_csv('directors.csv',index_col=0)
directors.head()
```

	director_name	id	gender
0	James Cameron	4762	Male
1	Gore Verbinski	4763	Male
2	Sam Mendes	4764	Male
3	Christopher Nolan	4765	Male
4	Andrew Stanton	4766	Male

```
directors.shape
```

```
(2349, 3)
```

```
directors["id"].nunique()
```

```
2349
```

```
movies["id"].nunique()
```

```
1465
```

```
movies.merge(directors, left_on="director_id", right_on="id").shape
```

```
(1465, 14)
```

```
movies["director_id"].isin(directors["id"])
```

```
0       True
1       True
2       True
3       True
4       True
        ...
1460    True
1461    True
1462    True
1463    True
1464    True
Name: director_id, Length: 1465, dtype: bool
```

```
np.all(movies["director_id"].isin(directors["id"]))
```

```
True
```

```
data = movies.merge(directors, how="left", left_on="director_id", right_on="id")
data.head()
```

	id_x	budget	popularity	revenue	title	vote_average	vote_count	director_id	year	month	day	director_n
0	43597	237000000	150	2787965087	Avatar	7.2	11800	4762	2009	Dec	Thursday	James Cam
1	43598	300000000	139	961000000	Pirates of the Caribbean: At World's End	6.9	4500	4763	2007	May	Saturday	Gore Verb
2	43599	245000000	107	880674609	Spectre	6.3	4466	4764	2015	Oct	Monday	Sam Me
					The Dark							Christo

```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 1465 entries, 0 to 1464
Data columns (total 14 columns):
 #   Column         Non-Null Count  Dtype  
---  ------         --------------  -----  
 0   id_x           1465 non-null   int64  
 1   budget         1465 non-null   int64  
 2   popularity     1465 non-null   int64  
 3   revenue        1465 non-null   int64  
 4   title          1465 non-null   object 
 5   vote_average   1465 non-null   float64
 6   vote_count     1465 non-null   int64  
 7   director_id    1465 non-null   int64  
 8   year           1465 non-null   int64  
 9   month          1465 non-null   object 
 10  day            1465 non-null   object 
 11  director_name  1465 non-null   object 
 12  id_y           1465 non-null   int64  
 13  gender         1341 non-null   object 
dtypes: float64(1), int64(8), object(5)
memory usage: 171.7+ KB
```

```
data.drop(["id_x", "id_y", "director_id"], axis=1, inplace=True)
data.head()
```

	budget	popularity	revenue	title	vote_average	vote_count	year	month	day	director_name	gender
0	237000000	150	2787965087	Avatar	7.2	11800	2009	Dec	Thursday	James Cameron	Male
1	300000000	139	961000000	Pirates of the Caribbean: At World's End	6.9	4500	2007	May	Saturday	Gore Verbinski	Male
2	245000000	107	880674609	Spectre	6.3	4466	2015	Oct	Monday	Sam Mendes	Male
3	250000000	112	1084939099	The Dark Knight Rises	7.6	9106	2012	Jul	Monday	Christopher Nolan	Male
4	258000000	115	890871626	Spider-Man 3	5.9	3576	2007	May	Tuesday	Sam Raimi	Male

```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 1465 entries, 0 to 1464
Data columns (total 11 columns):
 #   Column         Non-Null Count  Dtype  
---  ------         --------------  -----  
 0   budget         1465 non-null   int64  
 1   popularity     1465 non-null   int64  
 2   revenue        1465 non-null   int64  
 3   title          1465 non-null   object 
 4   vote_average   1465 non-null   float64
 5   vote_count     1465 non-null   int64  
 6   year           1465 non-null   int64  
 7   month          1465 non-null   object 
 8   day            1465 non-null   object 
 9   director_name  1465 non-null   object 
 10  gender         1341 non-null   object 
dtypes: float64(1), int64(5), object(5)
memory usage: 137.3+ KB
```

```
data.describe()
```

	budget	popularity	revenue	vote_average	vote_count	year
count	1.465000e+03	1465.000000	1.465000e+03	1465.000000	1465.000000	1465.000000
mean	4.802295e+07	30.855973	1.432539e+08	6.368191	1146.396587	2002.615017
std	4.935541e+07	34.845214	2.064918e+08	0.818033	1578.077438	8.680141
min	0.000000e+00	0.000000	0.000000e+00	3.000000	1.000000	1976.000000
25%	1.400000e+07	11.000000	1.738013e+07	5.900000	216.000000	1998.000000
50%	3.300000e+07	23.000000	7.578164e+07	6.400000	571.000000	2004.000000
75%	6.600000e+07	41.000000	1.792469e+08	6.900000	1387.000000	2009.000000
max	3.800000e+08	724.000000	2.787965e+09	8.300000	13752.000000	2016.000000

```
data.describe(include=object)
```

	title	month	day	director_name	gender
count	1465	1465	1465	1465	1341
unique	1465	12	7	199	2
top	Avatar	Dec	Friday	Steven Spielberg	Male
freq	1	193	654	26	1309

```
movies["director_id"].nunique()
```

```
199
```

```
data["revenue"] = (data["revenue"]/1000000).round(2)
```

```
data["budget"] = (data["budget"]/1000000).round(2)
```

```
data.head()
```

	budget	popularity	revenue	title	vote_average	vote_count	year	month	day	director_name	gender
0	237.0	150	2787.97	Avatar	7.2	11800	2009	Dec	Thursday	James Cameron	Male
1	300.0	139	961.00	Pirates of the Caribbean: At World's End	6.9	4500	2007	May	Saturday	Gore Verbinski	Male
2	245.0	107	880.67	Spectre	6.3	4466	2015	Oct	Monday	Sam Mendes	Male
3	250.0	112	1084.94	The Dark Knight Rises	7.6	9106	2012	Jul	Monday	Christopher Nolan	Male
4	258.0	115	890.87	Spider-Man 3	5.9	3576	2007	May	Tuesday	Sam Raimi	Male

```
# Querying the dataframe
```

```
data["vote_average"] >= 7
```

```
0        True
1       False
2       False
3        True
4       False
        ...
1460     True
1461     True
1462    False
1463    False
1464    False
Name: vote_average, Length: 1465, dtype: bool
```

```
data.loc[data["vote_average"] >= 7, :]
```

	budget	popularity	revenue	title	vote_average	vote_count	year	month	day	director_name	gender
0	237.00	150	2787.97	Avatar	7.2	11800	2009	Dec	Thursday	James Cameron	Male
3	250.00	112	1084.94	The Dark Knight Rises	7.6	9106	2012	Jul	Monday	Christopher Nolan	Male
8	200.00	145	1065.66	Pirates of the Caribbean: Dead Man's Chest	7.0	5246	2006	Jun	Tuesday	Gore Verbinski	Male
14	250.00	120	956.02	The Hobbit: The Battle of the Five Armies	7.1	4760	2014	Dec	Wednesday	Peter Jackson	Male
16	250.00	94	958.40	The Hobbit: The Desolation of Smaug	7.6	4524	2013	Dec	Wednesday	Peter Jackson	Male
...	...	...	...	...	...	...	...	...	...	...	...
1456	0.01	20	7.00	Eraserhead	7.5	485	1977	Mar	Saturday	David Lynch	Male
1457	0.00	5	0.00	The Mighty	7.1	51	1998	Oct	Friday	Peter Chelsom	Male
1458	0.06	27	3.22	Pi	7.1	586	1998	Jul	Friday	Darren Aronofsky	Male
1460	0.00	3	0.32	The Last Waltz	7.9	64	1978	May	Monday	Martin Scorsese	Male
1461	0.03	19	3.15	Clerks	7.4	755	1994	Sep	Tuesday	Kevin Smith	Male

363 rows × 11 columns

```
data[data["vote_average"] >= 7] # not recommened
# syntax not making clear whether it is using explcity or implcity infices
```

	budget	popularity	revenue	title	vote_average	vote_count	year	mo
0	237.00	150	2787.97	Avatar	7.2	11800	2009	
3	250.00	112	1084.94	The Dark Knight Rises	7.6	9106	2012	
8	200.00	145	1065.66	Pirates of the Caribbean: Dead Man's Chest	7.0	5246	2006	
14	250.00	120	956.02	The Hobbit: The Battle of the Five Armies	7.1	4760	2014	
16	250.00	94	958.40	The Hobbit: The Desolation of Smaug	7.6	4524	2013	
...	...	...	...	...	...	...	...	
1456	0.01	20	7.00	Eraserhead	7.5	485	1977	
1457	0.00	5	0.00	The Mighty	7.1	51	1998	
1458	0.06	27	3.22	Pi	7.1	586	1998	
1460	0.00	3	0.32	The Last Waltz	7.9	64	1978	
1461	0.03	19	3.15	Clerks	7.4	755	1994	

363 rows × 11 columns

```
# give title and director name for highly rated movies
data.loc[data["vote_average"] >= 7, ["title", "director_name"]]
```

	title	director_name
0	Avatar	James Cameron
3	The Dark Knight Rises	Christopher Nolan
8	Pirates of the Caribbean: Dead Man's Chest	Gore Verbinski
14	The Hobbit: The Battle of the Five Armies	Peter Jackson
16	The Hobbit: The Desolation of Smaug	Peter Jackson
...	...	...
1456	Eraserhead	David Lynch
1457	The Mighty	Peter Chelsom
1458	Pi	Darren Aronofsky
1460	The Last Waltz	Martin Scorsese
1461	Clerks	Kevin Smith

363 rows × 2 columns

```
# give title and director name for highly rated movies which were released in or after 2015
data.loc[(data["vote_average"]>=7) & (data["year"] >= 2015), ["title", "director_name"]]
```

	title	director_name
30	Furious 7	James Wan
78	Mad Max: Fury Road	George Miller
106	The Revenant	Alejandro González Iñárritu
162	The Martian	Ridley Scott
312	The Man from U.N.C.L.E.	Guy Ritchie
394	The Hateful Eight	Quentin Tarantino
519	13 Hours: The Secret Soldiers of Benghazi	Michael Bay
617	The Conjuring 2	James Wan
625	The Intern	Nancy Meyers
635	Bridge of Spies	Steven Spielberg
808	Southpaw	Antoine Fuqua
833	Straight Outta Compton	F. Gary Gray
839	The Big Short	Adam McKay
1344	Race	Stephen Hopkins

```
# Find me rows corresponding top 5 popular movies
data.sort_values(["popularity"], ascending=False).head(5)
```

	budget	popularity	revenue	title	vote_average	vote_count	year	mont
58	165.0	724	675.12	Interstellar	8.1	10867	2014	N
78	150.0	434	378.86	Mad Max: Fury Road	7.2	9427	2015	M
119	140.0	271	655.01	Pirates of the Caribbean: The Curse of the Bla...	7.5	6985	2003	J
120	125.0	206	752.10	The Hunger Games: Mockingjay - Part 1	6.6	5584	2014	N
45	185.0	187	1004.56	The Dark Knight	8.2	12002	2008	J

```
# title for all the movies directed by "Christopher Nolan"
data.loc[data["director_name"] == "Christopher Nolan", ["title"]]
```

```
# string methods --> learn later
```

	title
3	The Dark Knight Rises
45	The Dark Knight
58	Interstellar
59	Inception
74	Batman Begins
565	Insomnia
641	The Prestige
1341	Memento

```
# apply
# gender --> Male, Female (0, 1)
# ML Algos want all the features to be numerical
```

```
def encode(gender):
  if gender == "Male":
    return 0
  elif gender == "Female":
    return 1
  else:
    return gender
```

```
data["gender"] = data["gender"].apply(encode)
```

```
0       0
1       0
2       0
3       0
4       0
       ..
1460    0
1461    0
1462    0
1463    0
1464    1
Name: gender, Length: 1465, dtype: int64
```

```
# Fund sum of revenue and budget for every movie?
data["revenue"] + data["budget"]
```

```
0       3024.97
1       1261.00
2       1125.67
3       1334.94
4       1148.87
          ...
1460       0.32
1461       3.18
1462       0.00
1463       0.00
1464       2.26
Length: 1465, dtype: float64
```

```
data[["revenue", "budget"]].apply(np.sum, axis=1)
# apply can actually perform operations on both the axis
```

```
0       3024.97
1       1261.00
2       1125.67
3       1334.94
4       1148.87
          ...
1460       0.32
1461       3.18
1462       0.00
1463       0.00
1464       2.26
Length: 1465, dtype: float64
```

```
# the profit for every movie - (revenue - budget)
def calc_profit(x):
  return x["revenue"] - x["budget"]
data["profit"] = data[["revenue", "budget"]].apply(calc_profit, axis=1)
```

```
data
```

evenue	title	vote_average	vote_count	year	month	day	director_name	g
2787.97	Avatar	7.2	11800	2009	Dec	Thursday	James Cameron	
961.00	Pirates of the Caribbean: At World's End	6.9	4500	2007	May	Saturday	Gore Verbinski	
880.67	Spectre	6.3	4466	2015	Oct	Monday	Sam Mendes	
1084.94	The Dark Knight Rises	7.6	9106	2012	Jul	Monday	Christopher Nolan	
890.87	Spider-Man 3	5.9	3576	2007	May	Tuesday	Sam Raimi	
...	...	...	...	...	...	...	...	
0.32	The Last Waltz	7.9	64	1978	May	Monday	Martin Scorsese	
3.15	Clerks	7.4	755	1994	Sep	Tuesday	Kevin Smith	
0.00	Rampage	6.0	131	2009	Aug	Friday	Uwe Boll	
0.00	Slacker	6.4	77	1990	Jul	Friday	Richard Linklater	
2.04	El Mariachi	6.6	238	1992	Sep	Friday	Robert Rodriguez	

```
"ac15" > "ac14"
```

```
True
```