**1.ASK**

-What is the purpose of the analysis? Why did you choose IMDb?

- the analysis is a showcase to demonstrate my ability to extract insights and manipulate data as an -Entry level- business data analyst.

the chosen of the top 1000 movies of IMDB database because I it's kind of big data with various attributes which make it suitable for a showcase project.

**General Notes About the Data**: -

-Incomplete Data: Some fields, such as Certificate and Meta\_score, contain missing (NULL) values.

-The Gross (Revenue) column may not be available for all films.

-The formatting of some columns needs cleaning:

-Runtime: Stored as "142 min" (only numbers need to be extracted).

-Gross: Stored as "28,341,469"(Needs to remove "`28,341,469" (needs to remove "` and commas).)

-Genres: Movies of multiple genres are listed as comma-separated values ​​(e.g., "Drama, Crime").

-Timescale: Data includes films from 1921 ("The Kid") to 2020 ("Soul").

**2.prepare**

Data Source: [Kaggle](https://www.kaggle.com/datasets/harshitshankhdhar/imdb-dataset-of-top-1000-movies-and-tv-shows)

List of the top 1,000 movies as rated by IMDb.

-File Description:

-Number of rows: 1,000 (top 1,000 movies by rating).

-Number of columns: 16.

-Main columns (Poster\_Link, Series\_Title, Released\_Year, Certificate, Runtime, Genre, IMDB\_Rating, Overview, Meta\_score, Director, Star1, Star2, Star3, Star4, No\_of\_Votes & Gross)

**3. Process**

-Hiding “Poster\_Link” and “Overview” because no need for them in our investigation.

-looking for missing value in using filtering (found some value in "Certificate”,"Meta\_score","Gross"). No action is needed because if we deleted the movies missing “Gross” values will have an impact will aggregating “IMDb rating” over decades. On the other hand running the analysis with the missing values seemed to not have an effect.

-Checking the validity of the "IMDB\_Rating" sorting from largest to lowest gives us values under 10 specifically staring from 9.3.

- changing "Gross" data type to currency with 0 decimal places.

- splitting "Genre" column into 3 columns “Genre\_1 , Genre\_2, Genre\_3” to define main genre for movies.

**4. Analyze**

-using google “BigQuery” "SQL" to investigate the relationships between:

main actor appearances

AVG IMDB Rating for each genre

IMDB Rating for each decade

top 10 Director

total Gross of each genre

count total movies for each genre

MAX IMDB\_Rating

rating and gross correlation).

-using excel to manipulate data to group the “Released\_Year” into decades throw “LEN, CONCATE”.

-saving results as "CSV" file to import into " Microsoft Excel to make a plot later.

**- SQL code ( BigQuery)**

01-

SELECT (Star1) AS Actor,

count(Series\_Title) AS Number\_of\_movies

FROM `my-project-march-2025-453419.imdb\_top\_1000.imdb\_top\_1000`

WHERE Star1 IS NOT NULL AND Series\_Title IS NOT NULL

GROUP BY Star1

order by Number\_of\_movies DESC

LIMIT 10

02-

SELECT (Genre\_1) AS Main\_Genre,

ROUND(AVG(IMDB\_Rating),1) AS IMDB\_Rating

FROM `my-project-march-2025-453419.imdb\_top\_1000.imdb\_top\_1000`

WHERE Genre\_1 IS NOT NULL AND IMDB\_Rating IS NOT NULL

GROUP BY Genre\_1

order by IMDB\_Rating DESC

LIMIT 10

03-

SELECT (Director) AS Director,

count(Series\_Title) AS Number\_of\_movies

FROM `my-project-march-2025-453419.imdb\_top\_1000.imdb\_top\_1000`

WHERE Director IS NOT NULL AND Series\_Title IS NOT NULL

GROUP BY Director

order by Number\_of\_movies DESC

LIMIT 10

04-

SELECT (Genre\_1) AS Main\_Genre,

SUM(Gross) AS Gross\_of\_genre

FROM `my-project-march-2025-453419.imdb\_top\_1000.imdb\_top\_1000`

WHERE Genre\_1 IS NOT NULL AND Gross IS NOT NULL

GROUP BY Main\_Genre

order by Gross\_of\_genre DESC

05-

SELECT (Genre\_1) AS Main\_Genre,

count(Series\_Title) AS Number\_of\_movies

FROM `my-project-march-2025-453419.imdb\_top\_1000.imdb\_top\_1000`

WHERE Genre\_1 IS NOT NULL AND Series\_Title IS NOT NULL

GROUP BY Main\_Genre

order by Number\_of\_movies DESC

-06

SELECT (Genre\_1) AS Main\_Genre,

MAX(IMDB\_Rating) AS IMDB\_Rating

FROM `my-project-march-2025-453419.imdb\_top\_1000.imdb\_top\_1000`

WHERE Genre\_1 IS NOT NULL AND IMDB\_Rating IS NOT NULL

GROUP BY Genre\_1

order by IMDB\_Rating DESC

LIMIT 10

5. share

A presentation is made to demonstrate the different insights and the top three conclusion gained from the analysis of the database.