DATABASE MANAGEMENT AND DATABASE DESIGN
ASSIGNMENT 3
(Gathering, Scraping, Munging, and Cleaning Data)

Tv-show/Movie Recommendation System

TEAM MEMBERS -

Shreyas Rai (002769838)
Smiti Agrawal (002781419)
Sarthak Srivastava (002756847)

GITHUB LINK -

https://github.com/shreyashusky/tv_Shows_Recommendation_System.git
https://github.com/smitihusky/tvShowRecommendationSystem.git
https://github.com/sarthakhusky/TV-Shows-Recommendation-System.git

ABOUT

These days, the small screen has some very big things to offer. From sitcoms to dramas to travel and talk shows, all the small screen shows are the best in showing diversity.
TV shows are definitely one of the biggest business markets out there. Also, it offers an opportunity to upcoming artists. Over the years, the number of TV shows has increased exponentially and so has their customer base.

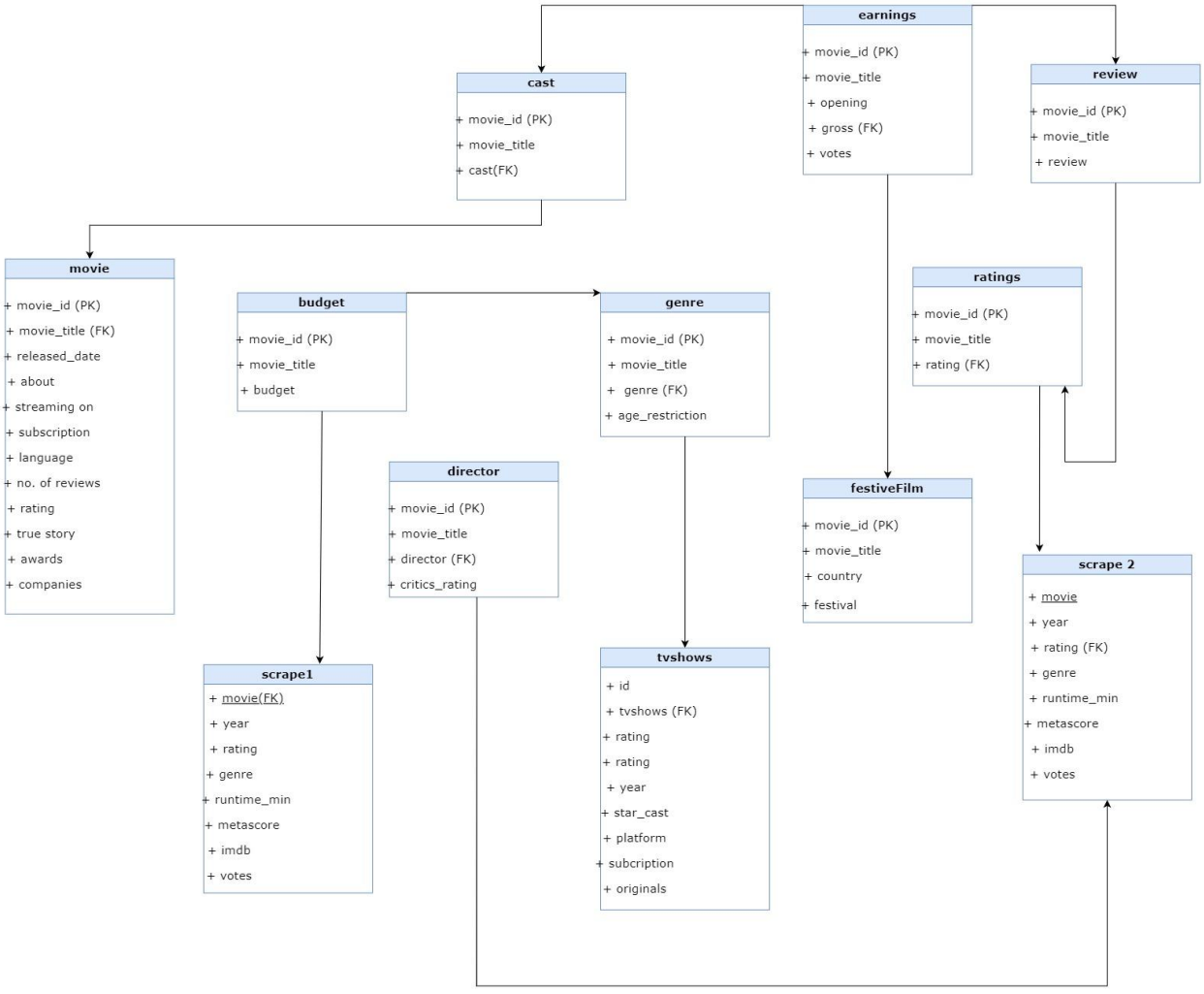
A survey conducted by www.deadline.com says that the top TV Shows can have 15-20 million viewership. With the advancement in technology and availability of cheap internet services, the reach of TV Shows is going to increase in the near future.

Our domain is Top TV Shows aired till date. We will use social networking APIs to get the data for entities which represent Companies, Genre, Subscription, Ratings, Producers and Consumers for our domain.

Entity Relationship DIAGRAM-

Explanation on some of the design decisions :

- We have collected data from web scraping(imdb) and kaggle. Where movie_id is a primary key.
- Each table has specific data which helps the user to find a particular move of his/her choice.
- In a movie table the movie_id is the primary key which helps to find the movies and avoid duplication.Where as movie_title is foreign key which is given as the name of the movie.
- A user can search a movie based on the directors, companies, age, festival, cast and county. Each movie will be distinguished and have a movie_id.
- Table connection helps the user to find the movies of different taste and genre and even cast and directors.
- Each data collected has uniformity and accuracy which help the user to find right movies .



SQL STATEMENTS-

BUDGET

```
CREATE TABLE "budget" (
        "movie_id"      INTEGER,
        "movie_title"   TEXT,
        "budget"        INTEGER
)
```

CAST

```
CREATE TABLE "cast" (
        "movie_id"      INTEGER,
        "movie_title"   TEXT,
        "cast"   TEXT
)
```

DIRECTOR

```
CREATE TABLE "director" (
        "movie_id"      INTEGER,
        "movie_title"   TEXT,
        "director"      TEXT,
        "critics_rating" INTEGER
)
```

EARNINGS

```
CREATE TABLE "earnings" (
        "movie_id"      INTEGER,
        "movie_title"   TEXT,
        "opening"       TEXT,
        "gross" INTEGER,
        "votes" INTEGER
)
```

FESTIVEFILM

```
CREATE TABLE "festiveFilm" (
        "movie_id"      INTEGER,
        "movie_title"   TEXT,
        "country"       TEXT,
        "festival"      TEXT
)
```

GENRE

```
CREATE TABLE "genre" (
```

```
	"movie_id"	INTEGER,
	"movie_title"	TEXT,
	"genre"TEXT,
	"certification"	TEXT
)
```

MOVIE

```
CREATE TABLE "movie" (
	"movie_id"	INTEGER,
	"movie_title"	TEXT,
	"released_date"		TEXT,
	"about"TEXT,
	"streamingon"	TEXT,
	"subscription"	TEXT,
	"language"	TEXT,
	"no.ofreviews"	INTEGER,
	"ratings"	INTEGER,
	"award_name"TEXT,
	"true_story"	TEXT,
	"companies"	TEXT
)
```

RATINGS

```
CREATE TABLE "ratings" (
	"movie_id"	INTEGER,
	"movie_title"	TEXT,
	"rating"INTEGER
)
```

REVIEW

```
CREATE TABLE "review" (
	"movie_id"	INTEGER,
	"movie_title"	TEXT,
	"review"	TEXT
)
```

SCRAPE1

```
CREATE TABLE "scrape1" (
	"field1"	INTEGER,
	"movie"	TEXT,
	"year"	INTEGER,
	"time_minute"	INTEGER,
	"imdb_rating"	REAL,
	"metascore"	TEXT,
```

```
        "vote"   INTEGER,
        "gross_earning"         TEXT
)
```

SCRAPE2

```
CREATE TABLE "scrape2" (
        "field1" INTEGER,
        "movie"         TEXT,
        "year"   INTEGER,
        "rating" TEXT,
        "genre"TEXT,
        "runtime_min" INTEGER,
        "imdb"  REAL,
        "metascore"     INTEGER,
        "votes" INTEGER,
        "n_imdb"        REAL
)
```

TVShows

```
CREATE TABLE "tvshow" (
        "id"      INTEGER,
        "tvshows"       TEXT,
        "rating" REAL,
        "year"   INTEGER,
        "star_cast"     TEXT,
        "field6" TEXT,
        "platform"      TEXT,
        "subscription"  TEXT,
        "originals"     TEXT
)
```

Inserting data in Table

```
for index,row in df_movie.iterrows():
cursor.execute("INSERT INTO Movies values (%s,%s,%s,%s)",
(int(row['movie_id']),row['movie_title'],row['about'],row['ratings']))
conn.commit()
```

SCRAPING DATA

In [5]:
```
# Storing each of the urls of 50 movies
for page in pages:
    # Getting the contents from the each url
    page = requests.get('https://www.imdb.com/search/title/?groups=top_1000&start=' + str(page) + '&ref_=adv_nxt', head
    soup = BeautifulSoup(page.text, 'html.parser')

    # Aiming the part of the html we want to get the information from
    movie_div = soup.find_all('div', class_='lister-item mode-advanced')

    # Controling the loop's rate by pausing the execution of the loop for a specified amount of time
    # Waiting time between requests for a number between 2-10 seconds
    sleep(randint(2,10))

    for container in movie_div:
        # Scraping the movie's name
        name = container.h3.a.text
        titles.append(name)

        # Scraping the movie's year
        year = container.h3.find('span', class_='lister-item-year').text
        years.append(year)

        # Scraping the movie's length
        runtime = container.find('span', class_='runtime').text if container.p.find('span', class_='runtime') else '-'
        time.append(runtime)

        # Scraping the rating
        imdb = float(container.strong.text)
        imdb_ratings.append(imdb)

        # Scraping the metascore
        m_score = container.find('span', class_='metascore').text if container.find('span', class_='metascore') else '-
        metascores.append(m_score)

        # Scraping votes and gross earnings
        nv = container.find_all('span', attrs={'name':'nv'})
```

CLEANING DATA

Cleaning of year column:

In [8]:
```
# Cleaning 'year' column
movies['year'] = movies['year'].str.extract('(\d+)').astype(int)
movies.head(3)
```

Out[8]:

	movie	year	time_minute	imdb_rating	metascore	vote	gross_earning
0	A Christmas Story	1983	93 min	7.9	77	151,583	$20.61M
1	Triangle of Sadness	2022	147 min	7.7	63	27,718	-
2	All Quiet on the Western Front	2022	148 min	7.8	76	74,549	-

In [9]:
```
# Cleaning 'time_minute' column
movies['time_minute'] = movies['time_minute'].str.extract('(\d+)').astype(int)
movies.head(3)
```

Out[9]:

	movie	year	time_minute	imdb_rating	metascore	vote	gross_earning
0	A Christmas Story	1983	93	7.9	77	151,583	$20.61M
1	Triangle of Sadness	2022	147	7.7	63	27,718	-
2	All Quiet on the Western Front	2022	148	7.8	76	74,549	-

In [10]:
```
# Cleaning 'metascore' column
movies['metascore'] = movies['metascore'].str.extract('(\d+)')
# convert it to float and if there are dashes turn it into NaN
movies['metascore'] = pd.to_numeric(movies['metascore'], errors='coerce')
```

In [11]:
```
# Cleaning 'vote' column
movies['vote'] = movies['vote'].str.replace(',', '').astype(int)
movies.head(3)
```

Out[11]:

	movie	year	time_minute	imdb_rating	metascore	vote	gross_earning
0	A Christmas Story	1983	93	7.9	77.0	151583	$20.61M

Cleaning of gross_earnings:

paste cells below

In [12]:
```
# Cleaning 'gross_earning' column
# left strip $ and right strip M
movies['gross_earning'] = movies['gross_earning'].map(lambda x: x.lstrip('$').rstrip('M'))
# convert it to float and if there are dashes turn it into NaN
movies['gross_earning'] = pd.to_numeric(movies['gross_earning'], errors='coerce')
movies.head(3)
```

Out[12]:

	movie	year	time_minute	imdb_rating	metascore	vote	gross_earning
0	A Christmas Story	1983	93	7.9	77.0	151583	20.61
1	Triangle of Sadness	2022	147	7.7	63.0	27718	NaN
2	All Quiet on the Western Front	2022	148	7.8	76.0	74549	NaN

Output:

paste cells below

	movie	year	time_minute	imdb_rating	metascore	vote	gross_earning
0	A Christmas Story	1983	93	7.9	77.0	151583	20.61
1	Triangle of Sadness	2022	147	7.7	63.0	27718	NaN
2	All Quiet on the Western Front	2022	148	7.8	76.0	74549	NaN
3	Top Gun: Maverick	2022	130	8.4	78.0	438268	NaN
4	Everything Everywhere All at Once	2022	139	8.1	81.0	244709	NaN
...	...	...	...	...	...	...	...
995	Airlift	2016	130	7.9	NaN	57326	NaN
996	Paan Singh Tomar	2012	135	8.2	NaN	36590	0.04
997	Anand	1971	122	8.1	NaN	34030	NaN
998	Sarfarosh	1999	174	8.1	NaN	25937	NaN
999	The Breath	2009	128	8.0	NaN	33972	NaN

1000 rows × 7 columns

In [15]:
```
movies.to_csv('c_movies.csv')
```

In [18]:
```
movies["gross_earning"]= movies["gross_earning"].fillna("Null")
```

5 USE-CASES (SMITI AGRAWAL)

**every use case includes joins of 2 tables

1. What are the family movies that are released during the Cannes film festival ?

```
In [104]: #UseCase
          #What are the family movies that are released during the Cannes film festival ?
          run_query('select g.movie_title,g.genre,f.festival
                    from genre g,festiveFilm f
                    where g.movie_id=f.movie_id and festival="Cannes" and genre like "%family%"')
```

Out[104]:

	movie_title	genre	festival
0	Tangled	Animation Family	Cannes
1	Alice in Wonderland	Family Fantasy Adventure	Cannes
2	Monsters University	Animation Family	Cannes
3	Cars 2	Animation Family Adventure Comedy	Cannes
4	Toy Story 3	Animation Family Comedy	Cannes
...	...	...	...
66	Fantasia 2000	Animation Family Music	Cannes
67	Looney Tunes: Back in Action	Animation Comedy Family	Cannes
68	Oceans	Documentary Family	Cannes
69	The Lion King	Family Animation Drama	Cannes
70	Cloudy with a Chance of Meatballs 2	Animation Family Comedy	Cannes

71 rows × 3 columns

2. What are the top 10 movies having higher profits?

```
In [107]: #What are the top 10 movies having higher profits?
          run_query('select e.movie_title,(e.gross-b.budget) as profit
                    from earnings e, budget b
                    where e.movie_id=b.movie_id
                    order by profit desc limit 10')
```

Out[107]:

	movie_title	profit
0	Avatar	2550965087
1	Titanic	1645034188
2	Jurassic World	1363528810
3	Furious 7	1316249360
4	The Avengers	1299557910
5	Avengers: Age of Ultron	1125403694
6	Frozen	1124219009
7	The Lord of the Rings: The Return of the King	1024888979
8	Iron Man 3	1015439994
9	Transformers: Dark of the Moon	928746996

3. Which director's movie has earned maximum money ?

```
In [116]: #Which director's movie has earned the maximum money ?
          run_query('select distinct d.director,sum(e.gross) as total_gross
                    from director d, earnings e
                    where d.movie_id=e.movie_id
                    group by d.director
                    order by e.gross desc limit 1')
```

Out[116]:

	director	total_gross
0	James cameran	2787965087

4. Highest rated movies released in the year 2000, USA?

```
In [133]: #Higheast rated movies released in year 2000, USA?
run_query('select m.movie_title,m.released_date, f.country, m.ratings
          from movie m,festiveFilm f
          where m.movie_id=f.movie_id and ratings>7 and m.released_date like "%2000%" and f.country like "%United State
          order by ratings desc')
```

Out[133]:

	movie_title	released_date	country	ratings
0	Gladiator	2000-05-01	United Kingdom United States of America	7.9
1	Cast Away	2000-12-22	United States of America	7.5
2	The Emperor's New Groove	2000-12-09	United States of America	7.2

5. What are real story based movies that have earned the highest votes?

```
In [142]: #What are real story based movies that has earned highest votes?
run_query('select distinct(m.movie_title), e.votes
          from movie m,earnings e where e.movie_id=m.movie_id and m.true_story="1"
          order by votes desc')
```

Out[142]:

	movie_title	votes
0	2012	4903
1	Zodiac	2023
2	Robin Hood	1398
3	Pompeii	1267
4	The Legend of Zorro	893
5	Gulliver's Travels	621
6	Ali	447
7	金陵十三釵	187
8	Town & Country	16

5 USE-CASES (SHREYAS RAI)

**every use case includes joins of 2 tables

1. What are top rated movies for children above age 13?

```
In [148]: #Top rated movies for children above age 13
run_query('select g.movie_title, g.certification, r.rating
          from genre g, ratings r
          where g.movie_id=r.movie_id and g.certification = "PG13"
          order by rating desc limit 20')
```

Out[148]:

	movie_title	certification	rating
0	The Dark Knight	PG13	8.2
1	Interstellar	PG13	8.1
2	Inception	PG13	8.1
3	The Lord of the Rings: The Return of the King	PG13	8.1
4	The Lord of the Rings: The Fellowship of the Ring	PG13	8.0
5	The Lord of the Rings: The Two Towers	PG13	8.0
6	Guardians of the Galaxy	PG13	7.9
7	Gladiator	PG13	7.9
8	The Wolf of Wall Street	PG13	7.9
9	The Departed	PG13	7.9
10	Shutter Island	PG13	7.8
11	Terminator 2: Judgment Day	PG13	7.7
12	The Dark Knight Rises	PG13	7.6
13	The Hobbit: The Desolation of Smaug	PG13	7.6
14	Edge of Tomorrow	PG13	7.6
15	Captain America: The Winter Soldier	PG13	7.6
16	The Martian	PG13	7.6
17	Finding Nemo	PG13	7.6

2. List award winning movies for Action packed

In [181]:
```
#List award winning movies for Action packed
run_query('select m.movie_title, g.genre, m.ratings
          from genre g, movie m
          where g.movie_id=m.movie_id
          and m.award_name = "1"
          and g.genre like "%Action%"
          order by m.ratings desc')
```

Out[181]:

	movie_title	genre	ratings
0	Inception	Action Thriller Science Fiction My...	8.1
1	The Lord of the Rings: The Return of the King	Adventure Fantasy Action	8.1
2	The Lord of the Rings: The Fellowship of the Ring	Adventure Fantasy Action	8.0
3	The Lord of the Rings: The Two Towers	Adventure Fantasy Action	8.0
4	Gladiator	Action Drama Adventure	7.9
5	Terminator 2: Judgment Day	Action Thriller Science Fiction	7.7
6	The Dark Knight Rises	Action Crime Drama Thriller	7.6
7	Edge of Tomorrow	Action Science Fiction	7.6
8	X-Men: Days of Future Past	Action Adventure Fantasy Science F...	7.5
9	Batman Begins	Action Crime Drama	7.5
10	The Avengers	Science Fiction Action Adventure	7.4
11	Star Trek Into Darkness	Action Adventure Science Fiction	7.4
12	The Hunger Games: Catching Fire	Adventure Action Science Fiction	7.4
13	Avatar	Action Adventure Fantasy Science F...	7.2

3. List the movies which has less ratings but earned more

In [160]:
```
#Movies which has less ratings but earned more
run_query('select e.movie_title
          from earnings e, ratings r
          where e.movie_id=r.movie_id
          and r.rating<5
          order by e.gross desc')
```

Out[160]:

	movie_title
0	Independence Day: Resurgence
1	The Last Airbender
2	Eragon
3	Batman & Robin
4	Gulliver's Travels
5	Fantastic Four
6	Speed 2: Cruise Control
7	Jack and Jill
8	Nutty Professor II: The Klumps
9	Cats & Dogs 2 : The Revenge of Kitty Galore
10	Catwoman
11	Stealth
12	How Do You Know
13	Battlefield Earth
14	Town & Country
15	The Adventures of Pluto Nash
16	A Sound of Thunder

4. Movies which were superhit in more than one country

In [177]:
```
#Movies which were superhit in more than one country
run_query('select r.movie_title as superhit_movies, f.country
          from ratings r, festiveFilm f
          where r.rating>7.1
          group by r.movie_title limit 10')
```

Out[177]:

	superhit_movies	country
0	A Beautiful Mind	United States of America United Kingdom
1	All That Jazz	United States of America United Kingdom
2	American Gangster	United States of America United Kingdom
3	Avatar	United States of America United Kingdom
4	Avengers: Age of Ultron	United States of America United Kingdom
5	Batman Begins	United States of America United Kingdom
6	Big Hero 6	United States of America United Kingdom
7	Black Hawk Down	United States of America United Kingdom
8	Blood Diamond	United States of America United Kingdom
9	Captain America: The Winter Soldier	United States of America United Kingdom

5. Cast and directors with maximum hit movies

In [170]:
```
#Cast and directors with maximum hit movies
run_query('select c.movie_title as Movie, d.director, c.cast
          from cast c, director d, ratings r
          where c.movie_id=d.movie_id
          and d.movie_id=r.movie_id
          and r.rating>8
          order by r.rating desc')
```

Out[170]:

	Movie	director	cast
0	The Dark Knight	Roberto Benigni	Bruce Wayne fecafd Christian Bale ...
1	Interstellar	Makoto Shinkai	Joseph Cooper febbfcecb Matthew McCon...
2	Inception	Nitesh Tiwari	Dom Cobb fecaede Leonardo DiCaprio ...
3	The Lord of the Rings: The Return of the King	Darius Marder	Frodo Baggins febcaff Elijah Wood ...

5 USE-CASES (SARTHAK SHRIVASTAV)

**every use case includes joins of 2 tables

1. What are the best movies available on Hulu?

In [197]:
```
#what are the Best movies avilable on Hulu?
run_query('select m.movie_title
          from review r, movie m
          where r.movie_title=m.movie_title and m.`streaming on` like "%hulu%" and r.review like "%Best%"')
```

Out[197]:

	movie_title
0	Avatar
1	The Hobbit: The Battle of the Five Armies
2	The Hobbit: The Desolation of Smaug
3	X-Men: Days of Future Past
4	Star Trek Into Darkness
5	The Great Gatsby
6	The Dark Knight
7	Edge of Tomorrow
8	Maleficent
9	Dawn of the Planet of the Apes
10	Captain America: The Winter Soldier
11	How to Train Your Dragon
12	Guardians of the Galaxy
13	The Curious Case of Benjamin Button
14	X-Men: First Class
15	Ratatouille
16	Batman Begins
17	Mad Max: Fury Road

2. Which directors has age restrictions as PG?

In [365]:
```
#which directors has age restriciton as PG
run_query('select d.movie_title,d.director
          from director d , genre g where d.movie_id=g.movie_id and g.certification = "PG"')
```

Out[365]:

	movie_title	director
0	Tangled	Nathan Greno
1	Harry Potter and the Half-Blood Prince	David Yates
2	The Chronicles of Narnia: Prince Caspian	Andrew Adamson
3	Monsters University	Dan Scanlon
4	Oz: The Great and Powerful	Sam Raimi
...	...	...
111	Arthur et les Minimoys	Luc Besson
112	Oceans	Gary Ross
113	Don Gato: El inicio de la pandilla	Andrés Couturier
114	The Lion King	Rob Minkoff
115	Journey 2: The Mysterious Island	Brad Peyton

3. What movies of directors are in the Berlin Festival?

In [370]:
```
#What movies of directors are in Berlin Festival?
run_query(' select d.movie_title, d.director, f.festival
          from director d , festiveFilm f
          where d.movie_id =f.movie_id and festival like "%Berlin%"')
```

Out[370]:

	movie_title	director	festival
0	Spectre	Sam mendes	Berlin
1	Harry Potter and the Half-Blood Prince	David Yates	Berlin
2	Quantum of Solace	Marc Forster	Berlin
3	Robin Hood	Otto Bathurst	Berlin
4	King Kong	Peter Jackson	Berlin
5	Skyfall	Sam Mendes	Berlin
6	X-Men: Days of Future Past	Steven Spielberg	Berlin
7	Up	David Fincher	Berlin
8	Monsters vs Aliens	Jonathan Demme	Berlin
9	Iron Man	George Lucas	Berlin
10	G.I. Joe: The Rise of Cobra	Martin Scorsese	Berlin
11	The Jungle Book	Ridley Scott	Berlin
12	Terminator 3: Rise of the Machines	Can Ulkay	Berlin
13	Inception	Nitesh Tiwari	Berlin
14	Alice Through the Looking Glass	Andrew Stanton	Berlin
15	Alexander	Stanley Kubrick	Berlin
16	Harry Potter and the Order of the Phoenix	Francis Ford Coppola	Berlin
17	Harry Potter and the Goblet of Fire	Ridley Scott	Berlin

4. Which actor were active in which year?

In [378]:
```
#Which actor were active in which year?
run_query('select m.movie_title, c.cast, m.released_date
          from movie m, cast c
          where m.movie_id=c.movie_id and c.cast like "%Robert Downey Jr.%"')
```

Out[378]:

	movie_title	cast	released_date
0	Avengers: Age of Ultron	Tony Stark / Iron Man edcde Robert Do...	2015-04-22
1	The Avengers	Tony Stark / Iron Man fecaeb Robert D...	2012-04-25
2	Captain America: Civil War	Steve Rogers / Captain America ecbeadb ...	2016-04-27
3	Iron Man 3	Tony Stark / Iron Man feacaedff Rober...	2013-04-18
4	Iron Man	Tony Stark / Iron Man fecafee Robert ...	2008-04-30
5	Iron Man 2	Tony Stark / Iron Man fec Robert Down...	2010-04-28
6	The Incredible Hulk	Bruce Banner / The Hulk fecafc Edward...	2008-06-12
7	Sherlock Holmes: A Game of Shadows	Sherlock Holmes fecafba Robert Downey...	2011-11-22
8	Tropic Thunder	Tugg Speedman fedcafe Ben Stiller ...	2008-08-09
9	Sherlock Holmes	Sherlock Holmes fecef Robert Downey J...	2009-12-23
10	Zodiac	Robert Graysmith fecafddd Jake Gyllen...	2007-03-02

\

5. Which actor have done a Romantic Movie?

```
In [380]: #Which actor have done a Romantic Movie?
          run_query('select c.cast from cast c,genre g where c.movie_id=g.movie_id and genre like "%Romance"')
```

Out[380]:

	cast
0	Jay Gatsby feecaea Leonardo DiCaprio ...
1	Prince Dastan fecafbe Jake Gyllenhaal...
2	Maleficent feaccafbcb Angelina Jolie ...
3	James Stewart / Jay Fennel feecae Jos...
4	Daisy feecafb Cate Blanchett Ben...
5	Alexander fecafed Colin Farrell ...
6	Isabella 'Bella' Swan fececaff Kriste...
7	Lisa Jorgenson fecafee Reese Withersp...
8	Porter Stoddard fecae Warren Beatty ...
9	Jack Byrnes feaca Robert De Niro ...
10	Frank Tupelo fec Johnny Depp Eli...
11	Carrie Bradshaw feecc Sarah Jessica P...
12	Giselle feccafff Amy Adams Rober...
13	Amanda Woods fecaf Cameron Diaz ...
14	Jane Adler fecaedc Meryl Streep ...
15	Sayuri fecafcf Zhang Ziyi Hatsum...
16	Michael Newman feecafbf Adam Sandler ...
17	Capt. Spurgeon 'Fish' Tanner febcafadf ...
18	Harry Sanborn febcaffc Jack Nicholson...
19	Skeeter Bronson fecad Adam Sandler ...

PREVIOUS ASSIGNMENTS USECASES
(Modeling 10 questions 10*3=30 questions)

- **What are the relevant shows for the given age group?**

```
In [200]: #What are the relevant shows for the given age group?
          run_query('select movie_title, certification from genre where certification="PG13" OR certification="PG"')
```

Out[200]:

	movie_title	certification
0	Avatar	PG13
1	Pirates of the Caribbean: At World's End	PG13
2	Spectre	PG13
3	The Dark Knight Rises	PG13
4	John Carter	PG13
...	...	...
459	Journey 2: The Mysterious Island	PG
460	Cloudy with a Chance of Meatballs 2	PG13
461	Red Dragon	PG13
462	Hidalgo	PG13
463	Jack and Jill	PG13

464 rows × 2 columns

- **What are the underrated shows to watch ?**

In [211]:
```
#What are the underrated show to watch ?
run_query('select movie_title from movie where ratings<7 and `no.of reviews`>5000')
```

Out[211]:

	movie_title
0	Batman v Superman: Dawn of Justice
1	Man of Steel
2	The Amazing Spider-Man
3	Jurassic World
4	Skyfall
5	Iron Man 3
6	World War Z
7	Suicide Squad
8	Iron Man 2
9	Thor
10	Spider-Man
11	Captain America: The First Avenger
12	The Hunger Games: Mockingjay - Part 1
13	Prometheus
14	The Hunger Games

- **Which is the cheapest platform to watch the show?**

In [216]:
```
#Which is the cheapest platform to watch the show?
run_query('select `streaming on` as cheapest_platform
          from movie
          group by `streaming on`
          order by cheapest_platform desc limit 1')
```

Out[216]:

	cheapest_platform
0	prime

- **What are the low budget- top rated shows?**

In [218]:
```
#What are the low budget- top rated show?
run_query('select r.movie_title from ratings r, budget b where r.movie_id=b.movie_id and rating>8 order by b.budget')
```

Out[218]:

	movie_title
0	The Lord of the Rings: The Return of the King
1	Inception
2	Interstellar
3	The Dark Knight

- **What are the shows that are popular in the country outside their origin country?**

In [222]:
```
#What are the shows that are popular in the country outside their origin country?
run_query('select f.movie_title, f.country from festiveFilm f, earnings e where f.movie_id=e.movie_id and votes>10000')
```

Out[222]:

	movie_title	country
0	Avatar	United States of America United Kingdom
1	The Avengers	United States of America
2	The Dark Knight	United Kingdom United States of America
3	Interstellar	Canada United States of America United...
4	Inception	United Kingdom United States of America
5	Django Unchained	United States of America

- **Who are involved in the cast and crew of the particular show?**

```
In [224]: #Who are involved in the cast and crew of the particular show?
          run_query('select tvshows,star_cast from tvshow')
```

Out[224]:

click to scroll output; double click to hide

	tvshows	star_cast
	rth II	David Attenborough, Chadden Hunter
1	Breaking Bad	Bryan Cranston, Aaron Paul
2	Planet Earth	Sigourney Weaver, David Attenborough
3	Band of Brothers	Scott Grimes, Damian Lewis
4	Chernobyl	Jessie Buckley, Jared Harris
...	...	...
95	Gullak	Jameel Khan, Geetanjali Kulkarni
96	Sarabhai V/S Sarabhai	Satish Shah, Ratna Pathak Shah
97	Invincible	Steven Yeun, J.K. Simmons
98	Panchayat	Jitendra Kumar, Raghuvir Yadav
99	Mad Men	Jon Hamm, Elisabeth Moss

100 rows × 2 columns

- **Which are the most viewed shows?**

```
In [229]: #Which are the most viewed shows?
          run_query('select movie_title,`no.of reviews` from movie where (`no.of reviews`)>10000')
```

Out[229]:

	movie_title	no.of reviews
0	Avatar	11800
1	The Avengers	11776
2	The Dark Knight	12002
3	Interstellar	10867
4	Inception	13752
5	Django Unchained	10099

- **Can a user filter the show by its durations?**

```
In [233]: #Can a user filter the show by its durations?
          run_query('select movie,time_minute as `duration(mins)` from scrape1 order by time_minute')
```

Out[233]:

click to scroll output; double click to hide

	movie	duration(mins)
	Sherlock Jr.	45
1	Freaks	64
2	Battleship Potemkin	66
3	The Cabinet of Dr. Caligari	67
4	The General	67
...	...	...
995	Once Upon a Time in America	229
996	Gone with the Wind	238
997	Zack Snyder's Justice League	242
998	Hamlet	242
999	Gangs of Wasseypur	321

1000 rows × 2 columns

- **What are the shows that are platform originals? Ex-shows that are netflix original.**

```
In [238]: #What are the shows that are platform originals? Ex-shows that are netflix original.
          run_query('select tvshows, platform from tvshow where originals="1"')
```

Out[238]:

	tvshows	platform
0	The Sopranos	HBO
1	Game of Thrones	HBO
2	Sherlock	BBC One
3	Batman: The Animated Series	prime
4	Scam 1992: The Harshad Mehta Story	sling
5	Fargo	hulu
6	TVF Pitchers	tvf
7	Black Mirror	None
8	Narcos	netflix
9	Dragon Ball Z	funimation
10	Dragon Ball Z	funimation
11	Six Feet Under	HBO
12	Kota Factory	netflix
13	Dark	netflix
14	The Boys	prime
15	Stranger Things	netflix
16	Gullak	sling
17	Panchayat	prime

- **Select top rated TV shows/movies based on their genre.**

```
In [241]: #Select top rated TV shows/movies based on their genre.
          run_query('select g.movie_title,
```

Out[241]:

	movie_title	genre
0	Titanic	Drama Romance Thriller
1	The Great Gatsby	Drama Romance
2	The Curious Case of Benjamin Button	Fantasy Drama Thriller Mystery ...
3	Cinderella Man	Romance Drama History
4	Memoirs of a Geisha	Drama History Romance
5	Déjà Vu	Romance Drama
6	A Beautiful Mind	Drama Romance

- **Select award winning TV shows/movies.**

```
In [242]: #Select award winning TV shows/movies.
          run_query('select movie_title from movie where award_name="1"')
```

Out[242]:

	movie_title
0	Avatar
1	The Dark Knight Rises
2	Tangled
3	Harry Potter and the Half-Blood Prince
4	The Avengers
5	The Hobbit: The Desolation of Smaug
6	Titanic
7	X-Men: Days of Future Past
8	Star Trek Into Darkness
9	Edge of Tomorrow
10	Inside Out
11	Interstellar
12	Inception
13	Harry Potter and the Goblet of Fire
14	Ratatouille
15	Batman Begins
16	The Hunger Games: Catching Fire
17	The Lord of the Rings: The Fellowship of the Ring
18	Gladiator

- **Select TV shows/movies based on real stories.**

```
In [245]: #Select TV shows/movies based on real stories.
run_query('select movie_title,released_date from movie where true_story="1"')
```

Out[245]:

	movie_title	released_date
0	Robin Hood	2010-05-12
1	2012	2009-10-10
2	Gulliver's Travels	2010-12-25
3	Ali	2001-12-11
4	Town & Country	2001-04-27
5	金陵十三釵	2011-12-15
6	Zodiac	2007-03-02
7	The Legend of Zorro	2005-10-24
8	Pompeii	2014-02-18

- **Select TV shows/movies according to Directors.**

```
In [247]: #Select TV shows/movies according to Directors.
run_query('select movie_title, director from director where director="Andrew Stanton"')
```

Out[247]:

	movie_title	director
0	John Carter	Andrew Stanton
1	Alice Through the Looking Glass	Andrew Stanton

- **Select a TV shows/movie which has sequence eg. part 1 or part 2**

```
In [255]: #Select a movie which has sequence eg. part 1 or part 2
run_query('select movie_title from movie where movie_title like "Spider-Man%"')
```

Out[255]:

	movie_title
0	Spider-Man 3
1	Spider-Man 2
2	Spider-Man

- **Select a TV show/movie which was released in more than one language/country.**

```
In [262]: #Select a TV show/movie which was released in more than one language/country.
run_query('select movie_title,country from festiveFilm where country like "%__________United States of America"')
```

Out[262]:

	movie_title	country
0	Spectre	United Kingdom United States of America
1	Harry Potter and the Half-Blood Prince	United Kingdom United States of America
2	Quantum of Solace	United Kingdom United States of America
3	Man of Steel	United Kingdom United States of America
4	The Chronicles of Narnia: Prince Caspian	CzCzech Republic Poland Slovenia U...
...	...	...
123	Nine	Italy United States of America
124	Red Planet	Australia United States of America
125	A Sound of Thunder	Czech Republic Germany United Kingdom...
126	Pompeii	Canada Germany United States of America
127	Red Dragon	Germany United States of America

128 rows × 2 columns

- **Select a TV show/movie based on the cast and crew.**

```
In [263]: #Select a TV show/movie based on the cast and crew.
          run_query('select * from cast')
```

Out[263]:

	movie_id	movie_title	cast
0	25683	Avatar	Jake Sully aacaca Sam Worthington ...
1	14531	Pirates of the Caribbean: At World's End	Captain Jack Sparrow fecafbd Johnny D...
2	18260	Spectre	James Bond fedcaeddb Daniel Craig ...
3	25056	The Dark Knight Rises	Bruce Wayne / Batman fecaf Christian ...
4	99197	John Carter	John Carter feacafea Taylor Kitsch ...
...	...	...	...
495	75730	Journey 2: The Mysterious Island	Hank fecaefe Dwayne Johnson Sean...
496	78945	Cloudy with a Chance of Meatballs 2	Flint Lockwood (voice) feabcafddf Bil...
497	57476	Red Dragon	Hannibal Lecter fecafbbd Anthony Hopk...
498	55309	Hidalgo	Frank Hopkins fedcaf Viggo Mortensen ...
499	72551	Jack and Jill	Jack / Jill fedcaef Adam Sandler ...

500 rows × 3 columns

- **Select a TV show/movie which was most watched on the released date.**

```
In [268]: #Select a TV show/movie which was most watched on the released date.
          run_query('select m.movie_title,m.released_date from movie m,earnings e where m.movie_id=e.movie_id and e.opening like
```

Out[268]:

	movie_title	released_date
0	Suicide Squad	2016-08-02
1	Jason Bourne	2016-07-27
2	Ghostbusters	2016-07-14
3	Star Trek Beyond	2016-07-07
4	The Legend of Tarzan	2016-06-29

- **What movies/TV shows require a subscription and on which platform?**

```
In [281]: #What movies/TV shows require a subscription and on which platform?
          run_query('select movie_title, `streaming on` from movie')
```

Out[281]:

click to scroll output; double click to hide

	movie_title	streaming on
0	Avatar	hulu
1	Pirates of the Caribbean: At World's End	hulu
2	Spectre	hulu
3	The Dark Knight Rises	netflix
4	John Carter	hulu
...	...	...
495	Journey 2: The Mysterious Island	hulu
496	Cloudy with a Chance of Meatballs 2	hulu
497	Red Dragon	hulu
498	Hidalgo	netflix
499	Jack and Jill	hulu

500 rows × 2 columns

- **What are the top critic's movies/tv-show?**

```
In [283]: #What are the top critic's movies/tv-show?
          run_query('select movie_title from director where critics_rating>7')
```

Out[283]:

	movie_title
0	Avatar
1	The Dark Knight Rises
2	Tangled
3	Avengers: Age of Ultron
4	Harry Potter and the Half-Blood Prince
...	...
94	Spirit: Stallion of the Cimarron
95	Déjà Vu
96	Oceans
97	A Beautiful Mind
98	The Lion King

99 rows × 1 columns

- **What are the top movies/tv-show from different Countries?**

```
In [288]: #What are the top movies/tv-show from different Countries?
          run_query('select movie_title, country from festiveFilm ')
```

Out[288]:

	movie_title	country
0	Avatar	United States of America United Kingdom
1	Pirates of the Caribbean: At World's End	United States of America
2	Spectre	United Kingdom United States of America
3	The Dark Knight Rises	United States of America
4	John Carter	United States of America
...	...	...
495	Journey 2: The Mysterious Island	United States of America
496	Cloudy with a Chance of Meatballs 2	United States of America
497	Red Dragon	Germany United States of America
498	Hidalgo	United States of America Morocco
499	Jack and Jill	United States of America

- **What are the different Companies(Warner Bros/ Phantom etc) and their top movies/Tv-show?**

```
In [293]: #What are the different Companies( Warner Bros/ Phantom etc)?
          run_query('select movie_title from movie where companies like "%Warner Bros.%"')
```

Out[293]:

	movie_title
0	The Dark Knight Rises
1	Harry Potter and the Half-Blood Prince
2	Batman v Superman: Dawn of Justice
3	Superman Returns
4	Man of Steel
...	...
81	Swordfish
82	Battlefield Earth
83	Looney Tunes: Back in Action
84	The Postman
85	Red Planet

86 rows × 1 columns

- **Which movies/Tv-show were released in theaters or had a platform streaming?**

```
In [297]: #Which movies/Tv-show were released on platform streaming?
          run_query('select movie_title from movie where `streaming on`= "netflix"')
```

Out[297]:

	movie_title
0	The Dark Knight Rises
1	Avengers: Age of Ultron
2	Harry Potter and the Half-Blood Prince
3	Pirates of the Caribbean: Dead Man's Chest
4	The Avengers
...	...
161	Babe: Pig in the City
162	Oceans
163	A Beautiful Mind
164	The Lion King
165	Hidalgo

166 rows × 1 columns

- **What are recently released movies/tv-shows?**

```
In [305]: #What are recently released movies/tv-shows?
          run_query('select movie_title, released_date from movie where released_date >2016-12-10 and released_date<2012-01-10')
```

Out[305]:

	movie_title	released_date
0	Titanic	1997-11-18
1	Wild Wild West	1999-06-29
2	Waterworld	1995-07-28
3	Armageddon	1998-07-01
4	Lethal Weapon 4	1998-07-10
...	...	...
71	Battlefield Earth	2000-05-10
72	The Postman	1997-12-25
73	Babe: Pig in the City	1998-11-25
74	Red Planet	2000-11-10
75	The Lion King	1994-06-23

76 rows × 2 columns

- **What are the top short films/Limited Series?**

```
In [314]: #What are the top short films/Limited Series?
          run_query('select movie, time_minute, imdb_rating from scrape1 where time_minute>60 and time_minute <99 order by imdb_ra
```

Out[314]:

	movie	time_minute	imdb_rating
0	12 Angry Men	96	9.0
1	The Lion King	88	8.5
2	Grave of the Fireflies	89	8.5
3	Modern Times	87	8.5
4	City Lights	87	8.5
...	...	...	...
165	The Station Agent	89	7.6
166	The Invisible Man	71	7.6
167	Philomena	98	7.6
168	Eyes Without a Face	90	7.6
169	The 39 Steps	86	7.6

170 rows × 3 columns

- **Which movies/tv-show were in the Film festival(Cannes)?**

```
In [315]: #Which movies/tv-show were in the Film festival(Cannes)?
          run_query('select movie_title from festiveFilm where festival = "Cannes"')
```

Out[315]:

	movie_title
0	Avatar
1	Pirates of the Caribbean: At World's End
2	John Carter
3	Tangled
4	Batman v Superman: Dawn of Justice
...	...
256	Oceans
257	A Sound of Thunder
258	Pompeii
259	The Lion King
260	Cloudy with a Chance of Meatballs 2

261 rows × 1 columns

- **Which movies/tv-show had a top box office collection?**

```
In [320]: #Which movies/tv-show had a top box office collection?
          run_query('select movie_title from budget where budget > 100000000')
```

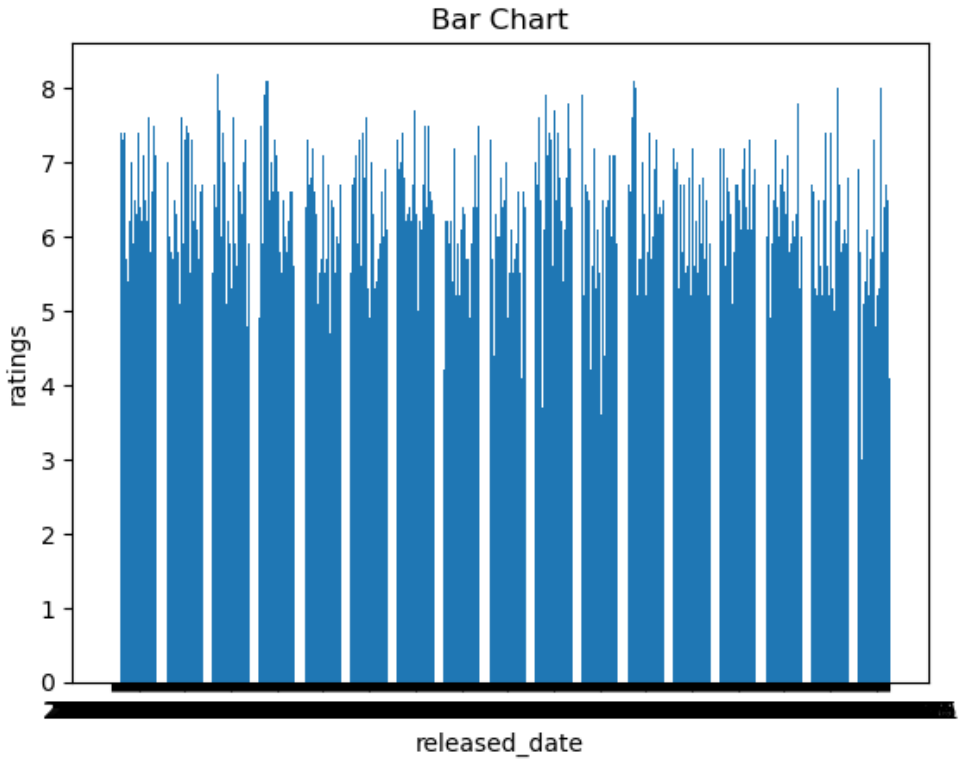
Out[320]:

	movie_title
0	Avatar
1	Pirates of the Caribbean: At World's End
2	Spectre
3	The Dark Knight Rises
4	John Carter
...	...
270	Lara Croft: Tomb Raider
271	The Alamo
272	Atlantis: The Lost Empire
273	Swordfish
274	Pompeii

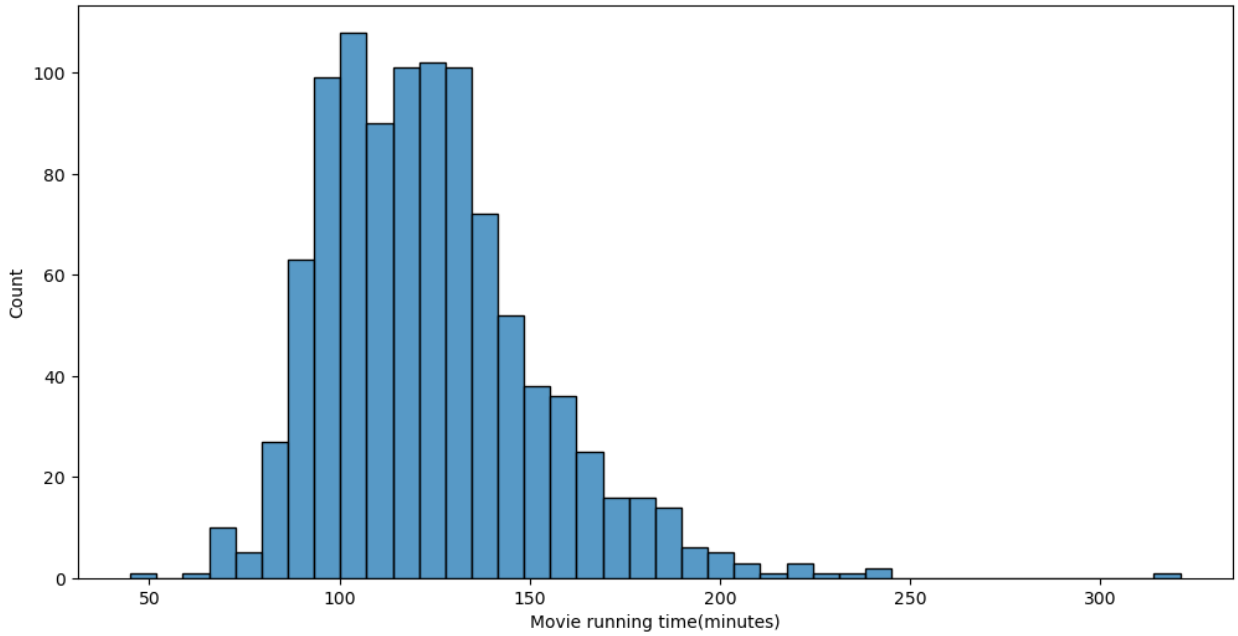
275 rows × 1 columns

DATA VISUALIZATION

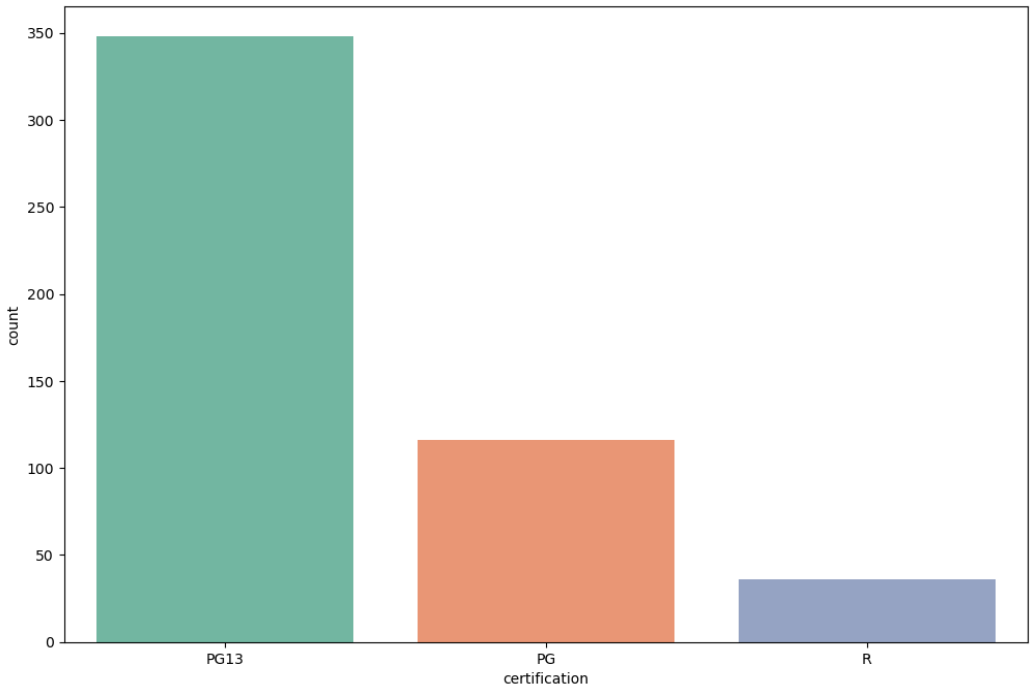
```
import pandas as pd
import matplotlib.pyplot as plt


# reading the database
data = pd.read_csv("movie.csv")

# Bar chart with day against tip
plt.bar(data['released_date'], data['ratings'])

plt.title("Bar Chart")

# Setting the X and Y labels
plt.xlabel('released_date')
plt.ylabel('ratings')

# Adding the legends
plt.show()
```



In [333]:
```
#Visualise the data according to Runtime ?
fig, ax = plt.subplots(figsize=(12,6))
sns.histplot(data=movie_df['time_minute'],ax=ax)
plt.xlabel('Movie running time(minutes)')
```

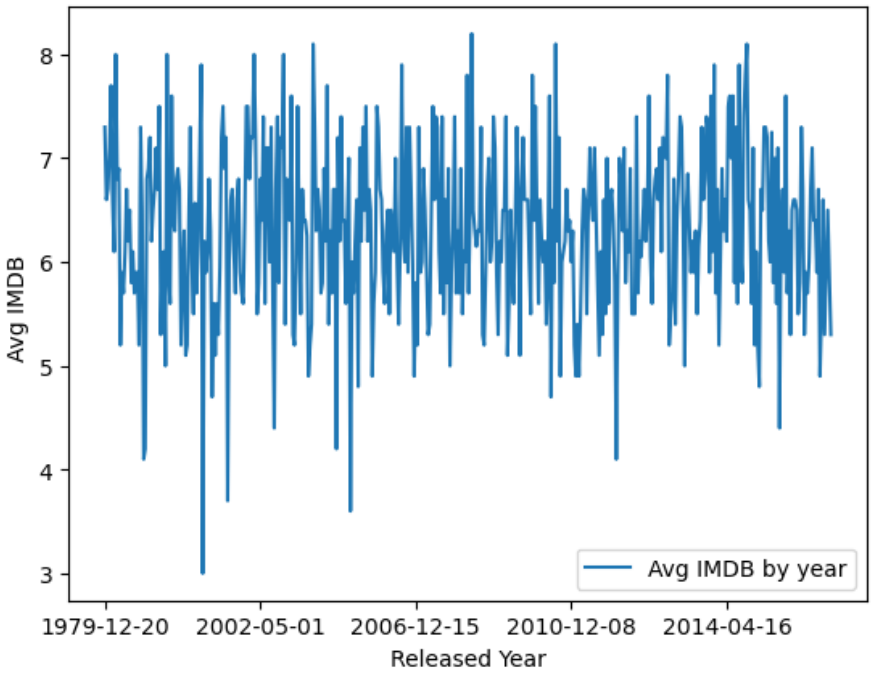


In [345]:
```
#Visualise the data according to the certification:
df2 = movie_df.groupby('certification')['movie_id'].mean()
fig, ax1 = plt.subplots(figsize=(12,8))
ax1 = sns.countplot(x="certification", data=movie_df, palette="Set2",
                    order=movie_df['certification'].value_counts().index[0:13])
```



click to scroll output; double click to hide

```
df1 = movie_df.groupby('released_date')['ratings'].mean()
df1.plot( label='Avg IMDB by year')
plt.legend()
plt.xlabel('Released Year')
plt.ylabel('Avg IMDB ')
plt.show()
```



In [364]:
```
# importing packages
import seaborn as sns
import matplotlib.pyplot as plt
import pandas as pd


# reading the database
data = pd.read_csv("movie.csv")

sns.histplot(x='ratings', data=data, kde=True, hue='streaming on')

plt.show()
```