University of Asia Pacific

Department of Computer Science & Engineering

Course Code: CSE - 212

Course Title: Database Systems Lab

Project Report

Project Name: Movie Recommendation Database Management System

Submitted By:

Name : Dipu Banik
Id: 22201255

Name : Sejon Hossain
Id: 22201272

Section : E-2
Semester : 2nd year 2nd semester

Submitted To:

Alif Ruslan, Lecturer
Department Of CSE

University of Asia Pacific.

Table Of Contents

❑ The topic that we have selected for our database project is Movie Recommendation Database Management System.

The **Movie Recommendation Database Management System** is designed to organize and manage information about movies, their genres, directors, actors, platforms, and awards, forming a robust foundation for building recommendation systems. By capturing extensive metadata, the database enables better movie discovery and personalized user experiences.

❑The ER diagram represents the database schema for a system that manages recommending movies. Below is a summarized explanation of its components and relationships:

❑ Entities and Attributes

1. Genres

This table represents movies by their genre.

- **Attributes :**
  - **genre_id** (Primary Key): Unique identifier for each genre.
  - genre_name: Name of the genre.

- **Sample Data:**
    - (1, 'Action');

---

2. Platforms

This table contains details about streaming platforms.

- **Attributes :**
    - platform_id (Primary Key): Unique identifier for each platform.
    - platform_name: Name of the platform.
    - subscription_cost: Monthly subscription cost in USD.
- **Sample Data:**
    - (1, 'Netflix', 15.99);

---

3. Awards

This table stores details of awards won by movies or individuals.

- **Attributes :**
    - award_id (Primary Key): Unique identifier for each award.
    - award_name: Name of the award.

- category: Specific category of the award.

- **Sample Data:**
  - (1, 'Oscar', 'Best Picture');

---

4. Directors
This table maintains information about movie directors.

- **Attributes:**
  - director_id (Primary Key): Unique identifier for each director.
  - name: Full name of the director.
  - birth_date: Date of birth.
  - nationality: Nationality of the director.
- **Sample Data:**
  - (1, 'Christopher Nolan', '1970-07-30', 'British');

---

4. Actors

This table stores details of actors involved in movies.

- **Attributes:**

  - actor_id (Primary Key): Unique identifier for each actor.
  - name: Full name of the actor.
  - birth_date: Date of birth.
  - nationality: Nationality of the actor.
  - total_movie: Total number of movies the actor has acted in.

- **Sample Data:**

  - (1, 'Leonardo DiCaprio', '1974-11-11', 'American', 30);

---

5. Critics

This table tracks movie critics and their details who usually rate movies.

- **Attributes:**

  - critic_id (Primary Key): Unique identifier for each critic.
  - critic_name: Name of the critic.
  - email: Contact email of the critic.
  - join_date: Date the critic joined the system.

- **Sample Data:**

  - (1, 'Roger Ebert', 'roger.ebert@example.com', '1995-06-15');

---

6. Movies

This table contains information about movies.

- **Attributes:**
  - **movie_id** (Primary Key): Unique identifier for each movie.
  - title: Title of the movie.
  - release_year: Year the movie was released.
  - duration: Duration of the movie.
  - **genre_id** (Foreign Key): Genre of the movie.
  - **director_id** (Foreign Key): Director of the movie.
  - language: Language of the movie.
  - industry: Industry the movie belongs to (e.g., Hollywood, Anime, Bollywood).

- **Sample Data:**
  - (1, 'Inception', 2010, '148 min', 6, 1, 'English', 'Hollywood');

8. Movie_Actors

This table maps actors to their respective roles in movies.

- **Attributes :**
  - movie_id (Foreign Key): ID of the movie.
  - actor_id (Foreign Key): ID of the actor.
  - role_name: Role played by the actor in the movie.
  - Primary key(movie_id, actor_id)

- **Sample Data :**
  - (1, 1, 'Dom Cobb') ;

9. Ratings

The table store ratings for movies which can be used to recommend or rank movies based on critic ratings.

- **Attributes :**
  - movie_id (Foreign Key): ID of the movie.
  - critic_id (Foreign Key): ID of the critic.
  - rating : Rating of each movie by critics .
  - rating_date : Date of the rating of movie.

    - Primary key(movie_id, critic_id)

- **Sample Data :**
    - (1, 1, 1, 9.0, '2010-07-16') ;

8. Movie_Awards

This table store data of movie who win awards and can use to recommend or rank based on awards.

- **Attributes :**
    - movie_id (Foreign Key): ID of the movie.
    - award_id (Foreign Key): ID of the award.
    - Primary key(movie_id, award_id)

- **Sample Data :**
    - (1, 6);

8. Movie_Platforms

This table maps actors to their respective roles in movies.

- **Attributes :**
  - movie_id (Foreign Key): ID of the movie.
  - platform_id (Foreign Key): ID of the actor.
  - abilability_date: Release date of movie on platforms.
  - Primary key(movie_id, platform_id)

- **Sample Data :**
  - (1, 1, '2010-09-01') ;

❑Relations

Entities and All Relationships

1. Genres and Movies

- **Relationship:** One-to-Many
- **Description:** Each movie belongs to a single genre (genre_id in Movies references Genres), but a genre can have many movies.

---

2. Directors and Movies

- **Relationship:** One-to-Many

- **Description:** Each movie is directed by one director (director_id in Movies references Directors), but a director can direct multiple movies.

---

3. Actors and Movies (via Movie_Actors)

- **Relationship:** Many-to-Many
- **Description:** A movie can feature multiple actors, and an actor can appear in multiple movies. The Movie_Actors table handles this relationship with additional role information.

---

4. Movies and Critics (via Ratings)

- **Relationship:** Many-to-Many (with Ratings)
- **Description:** Critics can rate and review multiple movies and a movie can also have multiple ratings by different critics. The Ratings table links Movies and Critics.

---

5. Awards and Movies (via Movie_Awards)

- **Relationship:** Many-to-Many
- **Description:** A movie can win multiple awards, and an award can be given to multiple movies across years and categories. This is captured in the Movie_Awards table.

---

6. Platforms and Movies (via Movie_Platforms)

- **Relationship:** Many-to-Many

- **Description:** A movie can be available on multiple platforms, and a platform can host multiple movies. The Movie_Platforms table tracks this along with the availability date.

---

7. Critics and Ratings

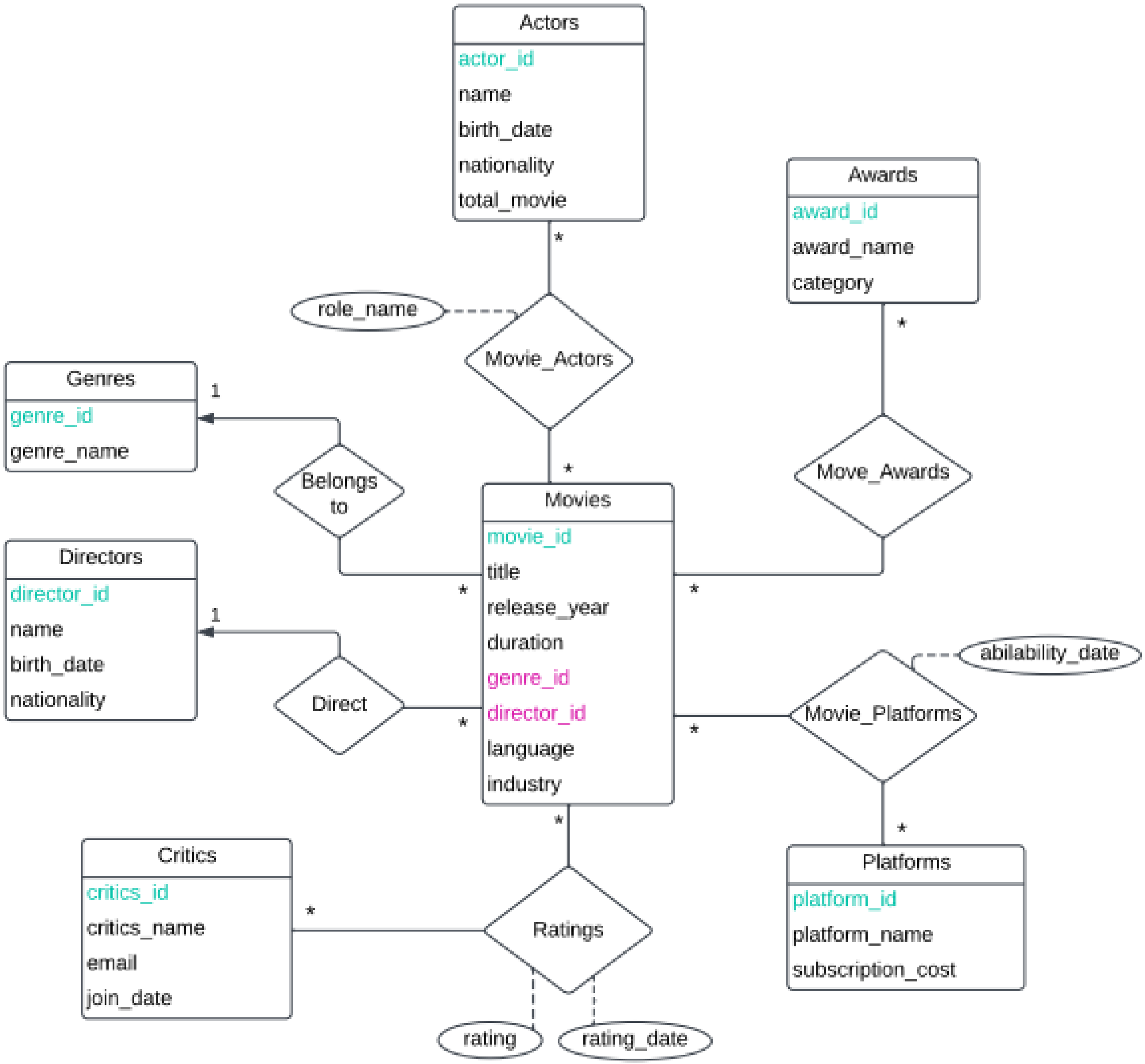
- **Relationship:** One-to-Many
- **Description:** Each critic can write multiple ratings/reviews (critic_id in Ratings references Critics), but each rating is associated with a single critic.

---

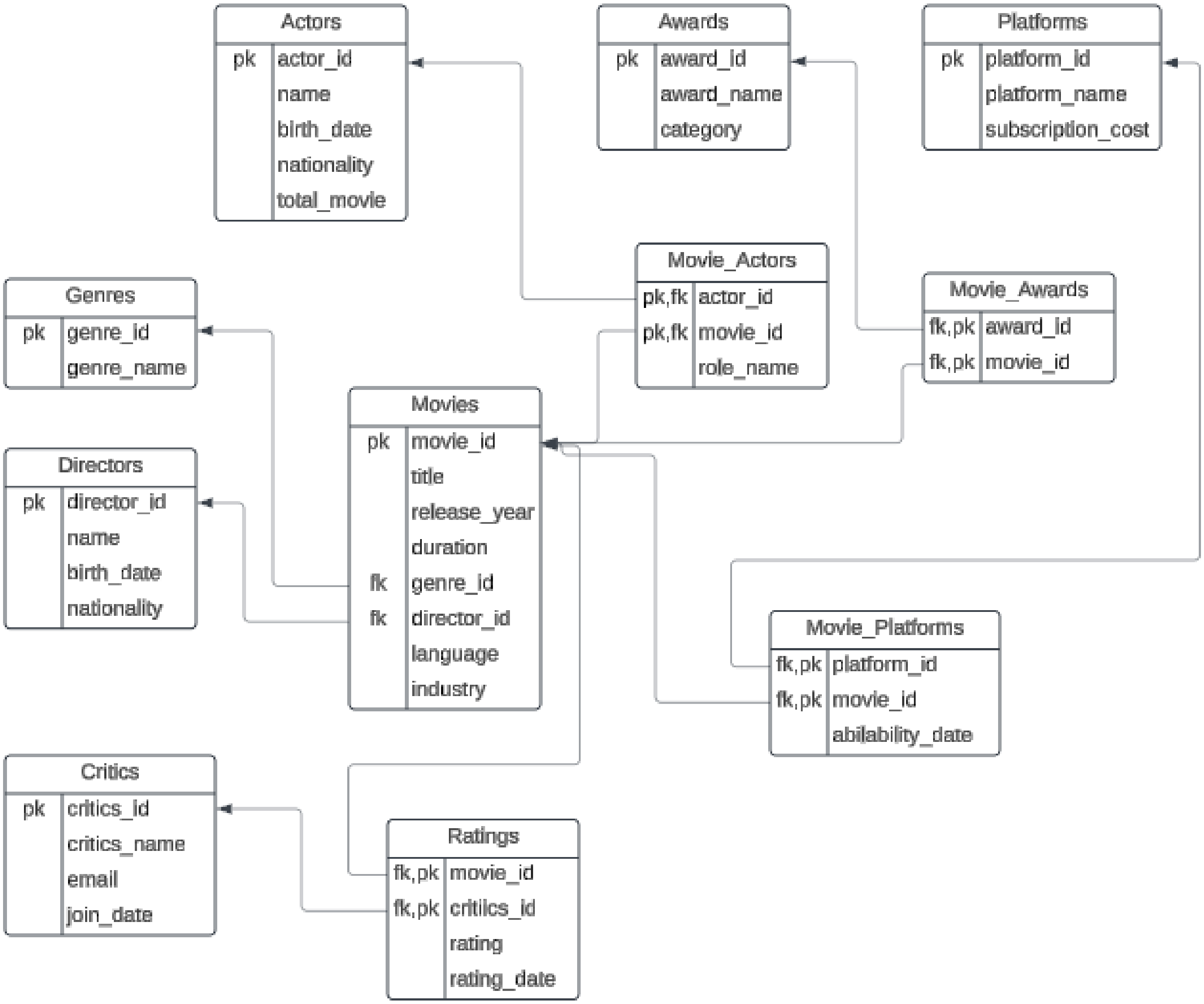
8. Summary

- **Core Table:** Movies
  - All other tables (e.g., Genres, Directors, Actors, Awards, Platforms, and Critics) relate to Movies either directly or via linking tables (Movie_Actors, Movie_Platforms, Movie_Awards, Ratings).

❑ER Diagram of Movie Recommendation Database Management System :



Schema diagram of Movie Recommendation Database Management System:



	Actors
pk	actor_id
	name
	birth_date
	nationality
	total_movie

	Awards
pk	award_id
	award_name
	category

	Platforms
pk	platform_id
	platform_name
	subscription_cost

	Movie_Actors
pk,fk	actor_id
pk,fk	movie_id
	role_name

	Movie_Awards
fk,pk	award_id
fk,pk	movie_id

	Genres
pk	genre_id
	genre_name

	Movies
pk	movie_id
	title
	release_year
	duration
fk	genre_id
fk	director_id
	language
	industry

	Directors
pk	director_id
	name
	birth_date
	nationality

	Movie_Platforms
fk,pk	platform_id
fk,pk	movie_id
	abilability_date

	Critics
pk	critics_id
	critics_name
	email
	join_date

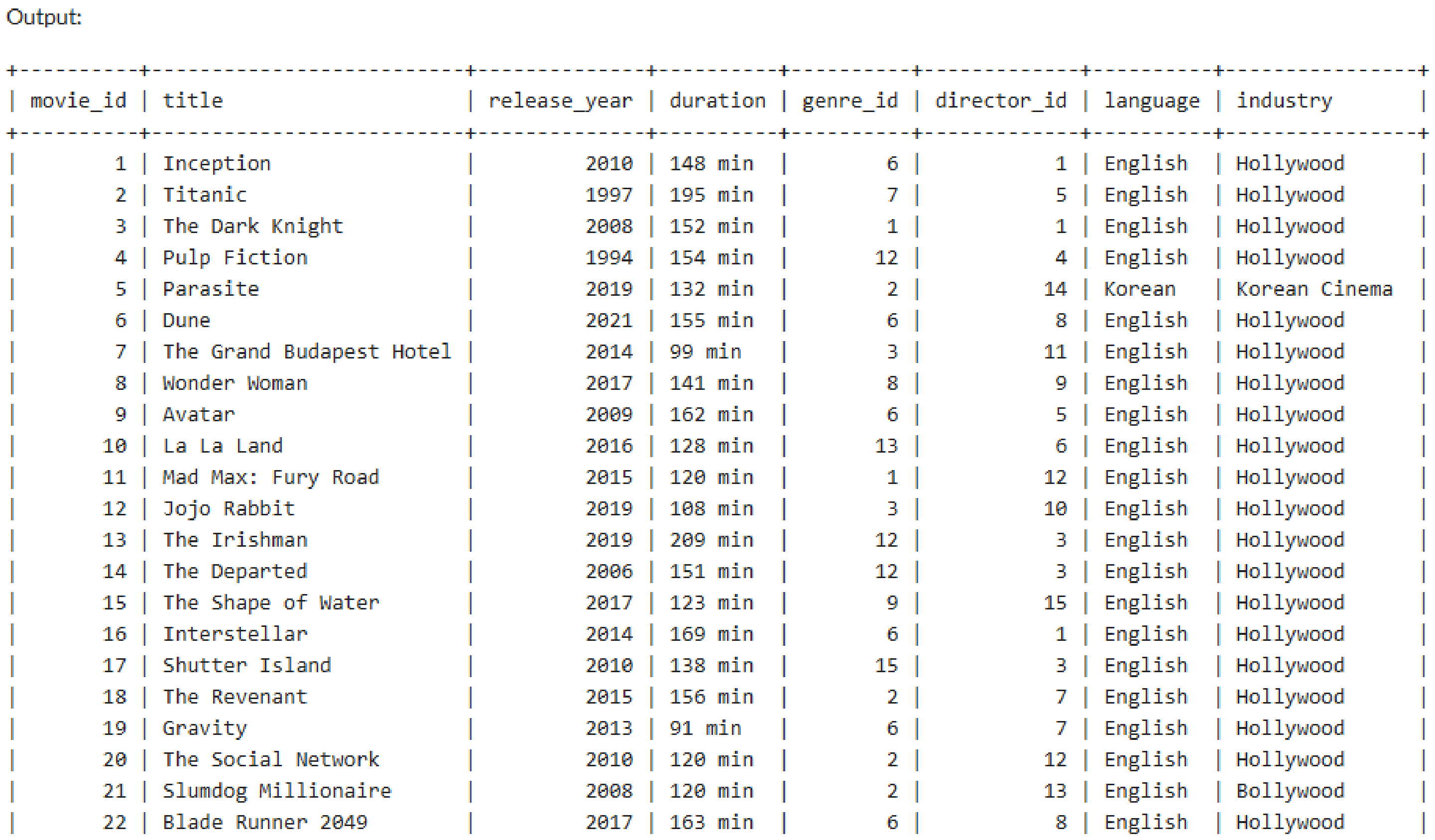
	Ratings
fk,pk	movie_id
fk,pk	critilcs_id
	rating
	rating_date

SQL Queries for the database –

(Basic Queries)

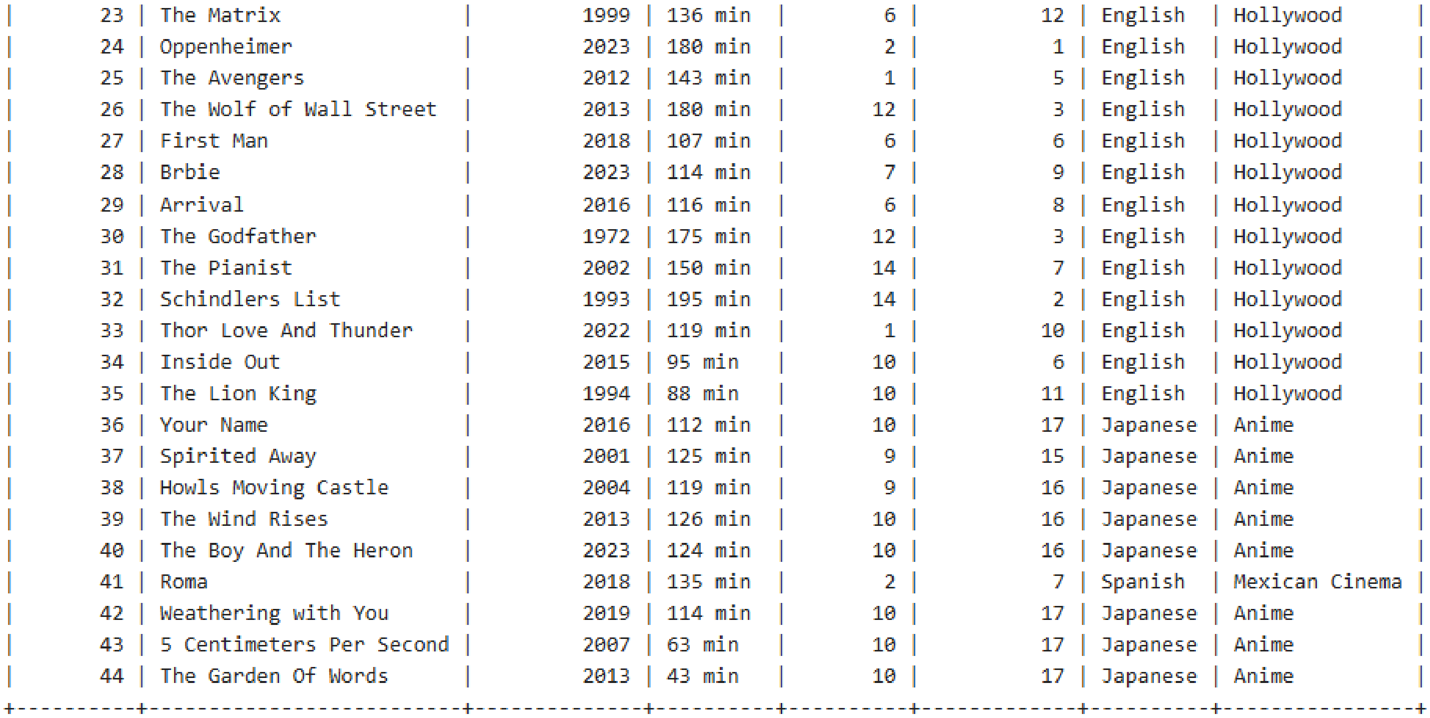
--1. Display all Movies information from the database.

select*from Movies;



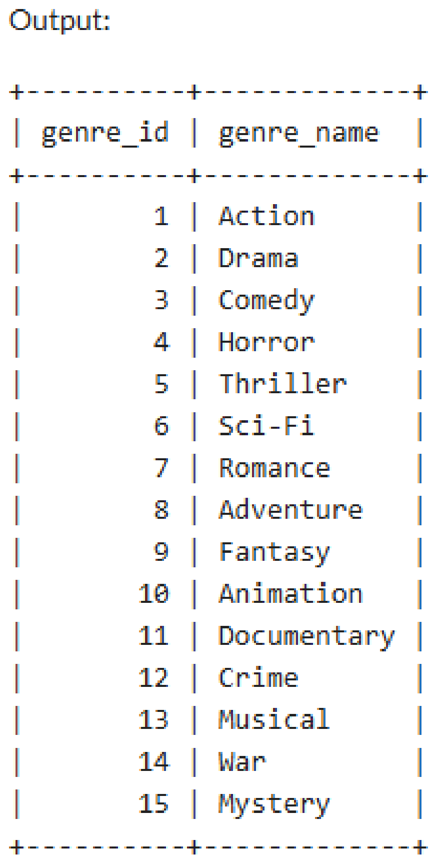
Output:

```
+----------+-------------------------+--------------+----------+----------+-------------+----------+----------------+
| movie_id | title                   | release_year | duration | genre_id | director_id | language | industry       |
+----------+-------------------------+--------------+----------+----------+-------------+----------+----------------+
|        1 | Inception               |         2010 | 148 min  |        6 |           1 | English  | Hollywood      |
|        2 | Titanic                 |         1997 | 195 min  |        7 |           5 | English  | Hollywood      |
|        3 | The Dark Knight         |         2008 | 152 min  |        1 |           1 | English  | Hollywood      |
|        4 | Pulp Fiction            |         1994 | 154 min  |       12 |           4 | English  | Hollywood      |
|        5 | Parasite                |         2019 | 132 min  |        2 |          14 | Korean   | Korean Cinema  |
|        6 | Dune                    |         2021 | 155 min  |        6 |           8 | English  | Hollywood      |
|        7 | The Grand Budapest Hotel |        2014 | 99 min   |        3 |          11 | English  | Hollywood      |
|        8 | Wonder Woman            |         2017 | 141 min  |        8 |           9 | English  | Hollywood      |
|        9 | Avatar                  |         2009 | 162 min  |        6 |           5 | English  | Hollywood      |
|       10 | La La Land              |         2016 | 128 min  |       13 |           6 | English  | Hollywood      |
|       11 | Mad Max: Fury Road      |         2015 | 120 min  |        1 |          12 | English  | Hollywood      |
|       12 | Jojo Rabbit             |         2019 | 108 min  |        3 |          10 | English  | Hollywood      |
|       13 | The Irishman            |         2019 | 209 min  |       12 |           3 | English  | Hollywood      |
|       14 | The Departed            |         2006 | 151 min  |       12 |           3 | English  | Hollywood      |
|       15 | The Shape of Water      |         2017 | 123 min  |        9 |          15 | English  | Hollywood      |
|       16 | Interstellar            |         2014 | 169 min  |        6 |           1 | English  | Hollywood      |
|       17 | Shutter Island          |         2010 | 138 min  |       15 |           3 | English  | Hollywood      |
|       18 | The Revenant            |         2015 | 156 min  |        2 |           7 | English  | Hollywood      |
|       19 | Gravity                 |         2013 | 91 min   |        6 |           7 | English  | Hollywood      |
|       20 | The Social Network      |         2010 | 120 min  |        2 |          12 | English  | Hollywood      |
|       21 | Slumdog Millionaire     |         2008 | 120 min  |        2 |          13 | English  | Bollywood      |
|       22 | Blade Runner 2049       |         2017 | 163 min  |        6 |           8 | English  | Hollywood      |
```



```
|       23 | The Matrix               |         1999 | 136 min  |        6 |          12 | English  | Hollywood      |
|       24 | Oppenheimer              |         2023 | 180 min  |        2 |           1 | English  | Hollywood      |
|       25 | The Avengers             |         2012 | 143 min  |        1 |           5 | English  | Hollywood      |
|       26 | The Wolf of Wall Street  |         2013 | 180 min  |       12 |           3 | English  | Hollywood      |
|       27 | First Man                |         2018 | 107 min  |        6 |           6 | English  | Hollywood      |
|       28 | Brbie                    |         2023 | 114 min  |        7 |           9 | English  | Hollywood      |
|       29 | Arrival                  |         2016 | 116 min  |        6 |           8 | English  | Hollywood      |
|       30 | The Godfather            |         1972 | 175 min  |       12 |           3 | English  | Hollywood      |
|       31 | The Pianist              |         2002 | 150 min  |       14 |           7 | English  | Hollywood      |
|       32 | Schindlers List          |         1993 | 195 min  |       14 |           2 | English  | Hollywood      |
|       33 | Thor Love And Thunder    |         2022 | 119 min  |        1 |          10 | English  | Hollywood      |
|       34 | Inside Out               |         2015 | 95 min   |       10 |           6 | English  | Hollywood      |
|       35 | The Lion King            |         1994 | 88 min   |       10 |          11 | English  | Hollywood      |
|       36 | Your Name                |         2016 | 112 min  |       10 |          17 | Japanese | Anime          |
|       37 | Spirited Away            |         2001 | 125 min  |        9 |          15 | Japanese | Anime          |
|       38 | Howls Moving Castle      |         2004 | 119 min  |        9 |          16 | Japanese | Anime          |
|       39 | The Wind Rises           |         2013 | 126 min  |       10 |          16 | Japanese | Anime          |
|       40 | The Boy And The Heron    |         2023 | 124 min  |       10 |          16 | Japanese | Anime          |
|       41 | Roma                     |         2018 | 135 min  |        2 |           7 | Spanish  | Mexican Cinema |
|       42 | Weathering with You      |         2019 | 114 min  |       10 |          17 | Japanese | Anime          |
|       43 | 5 Centimeters Per Second |         2007 | 63 min   |       10 |          17 | Japanese | Anime          |
|       44 | The Garden Of Words      |         2013 | 43 min   |       10 |          17 | Japanese | Anime          |
+----------+--------------------------+--------------+----------+----------+-------------+----------+----------------+
--2. Display all Genres information from the database.
```

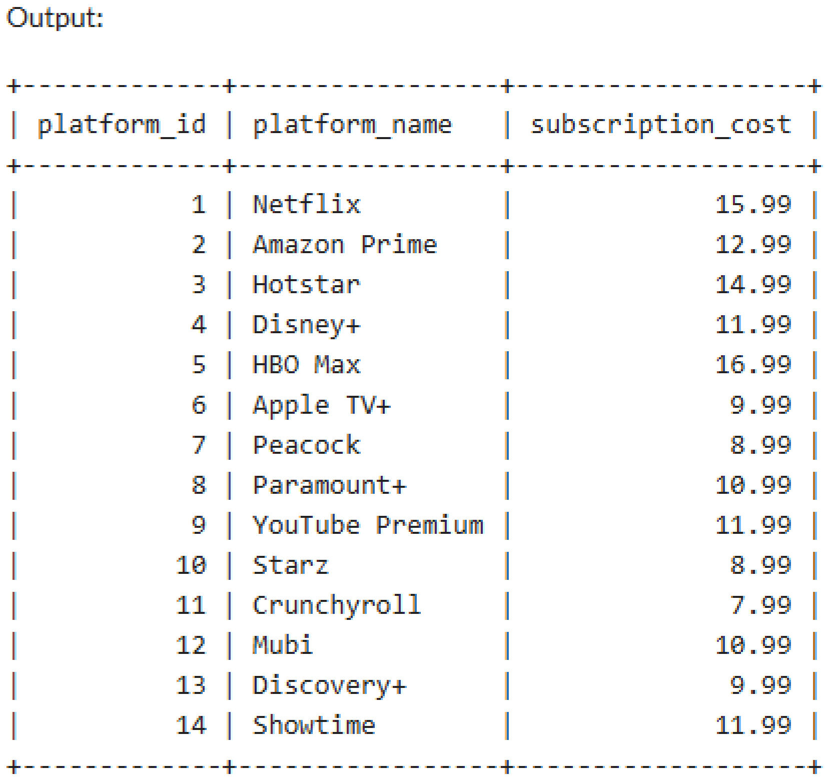
```
select*from Genres;
```



Output:

```
+----------+-------------+
| genre_id | genre_name  |
+----------+-------------+
|        1 | Action      |
|        2 | Drama       |
|        3 | Comedy      |
|        4 | Horror      |
|        5 | Thriller    |
|        6 | Sci-Fi      |
|        7 | Romance     |
|        8 | Adventure   |
|        9 | Fantasy     |
|       10 | Animation   |
|       11 | Documentary |
|       12 | Crime       |
|       13 | Musical     |
|       14 | War         |
|       15 | Mystery     |
+----------+-------------+
--3. Display all Platforms information from the
database.
```

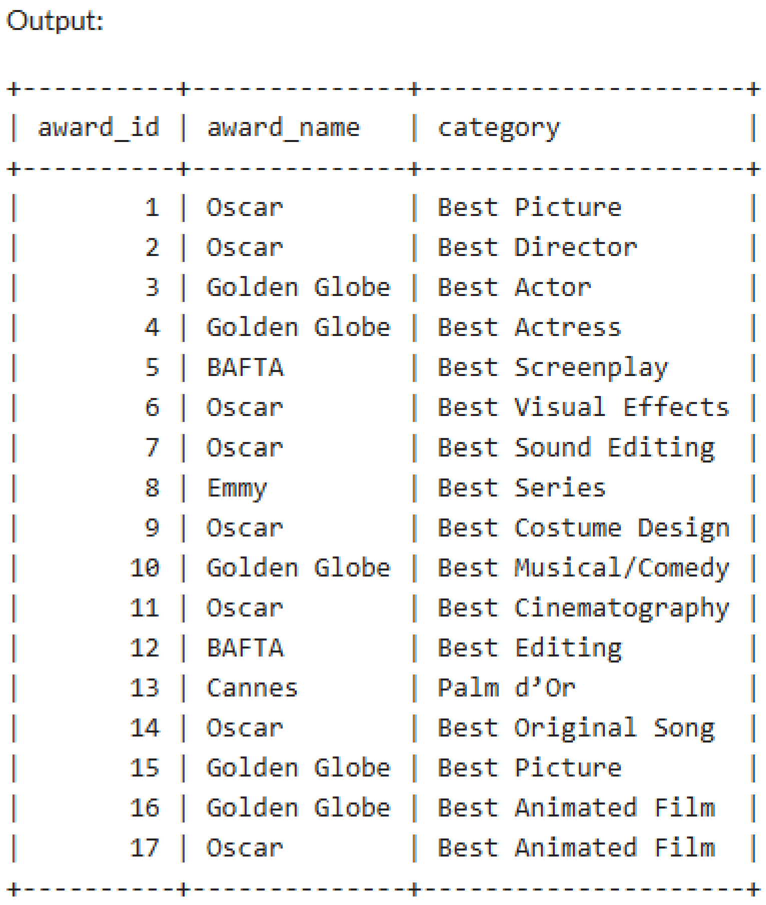
select*from Platforms;



Output:

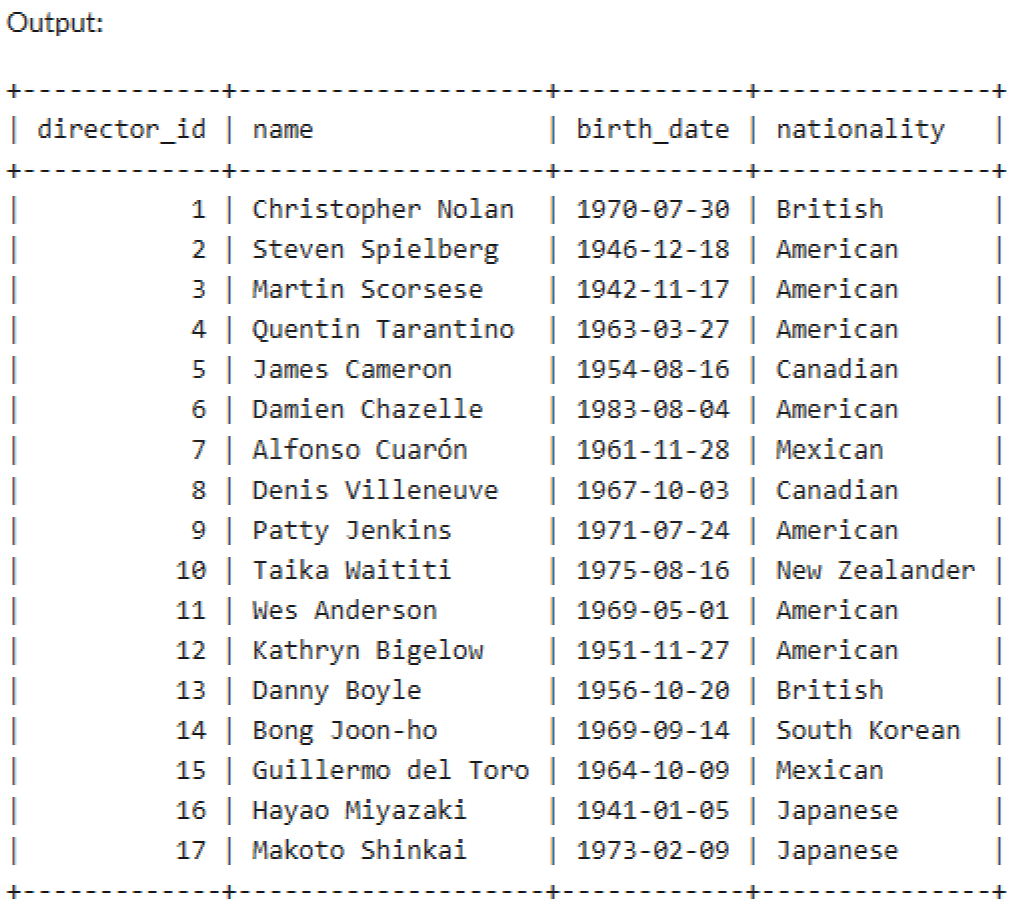
```
+-------------+-----------------+-------------------+
| platform_id | platform_name   | subscription_cost |
+-------------+-----------------+-------------------+
|           1 | Netflix         |             15.99 |
|           2 | Amazon Prime    |             12.99 |
|           3 | Hotstar         |             14.99 |
|           4 | Disney+         |             11.99 |
|           5 | HBO Max         |             16.99 |
|           6 | Apple TV+       |              9.99 |
|           7 | Peacock         |              8.99 |
|           8 | Paramount+      |             10.99 |
|           9 | YouTube Premium |             11.99 |
|          10 | Starz           |              8.99 |
|          11 | Crunchyroll     |              7.99 |
|          12 | Mubi            |             10.99 |
|          13 | Discovery+      |              9.99 |
|          14 | Showtime        |             11.99 |
+-------------+-----------------+-------------------+
--4. Display all Awards information from the
database.
```

select*from Awards;



Output:

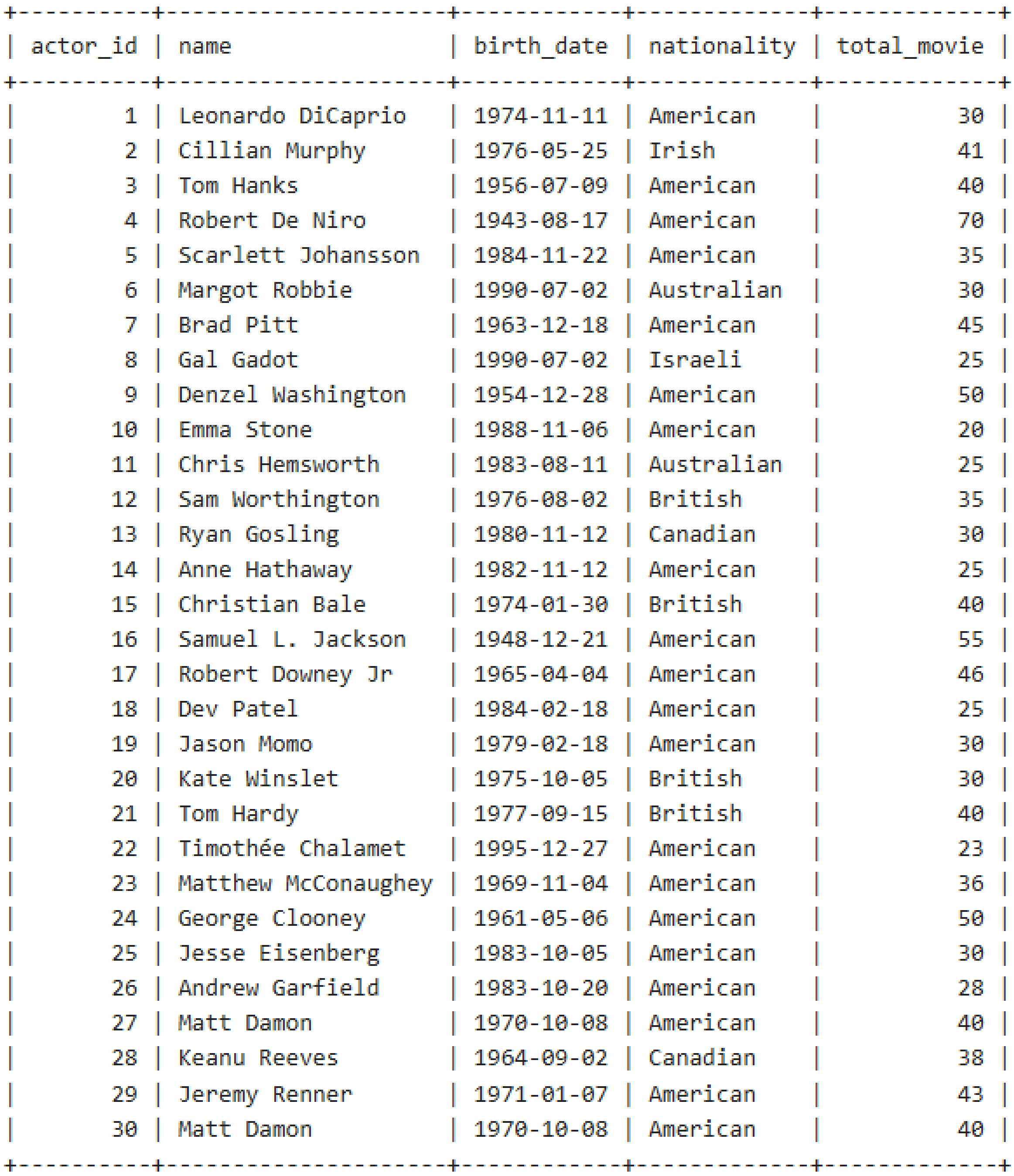
```
+----------+--------------+---------------------+
| award_id | award_name   | category            |
+----------+--------------+---------------------+
|        1 | Oscar        | Best Picture        |
|        2 | Oscar        | Best Director       |
|        3 | Golden Globe | Best Actor          |
|        4 | Golden Globe | Best Actress        |
|        5 | BAFTA        | Best Screenplay     |
|        6 | Oscar        | Best Visual Effects |
|        7 | Oscar        | Best Sound Editing  |
|        8 | Emmy         | Best Series         |
|        9 | Oscar        | Best Costume Design |
|       10 | Golden Globe | Best Musical/Comedy |
|       11 | Oscar        | Best Cinematography |
|       12 | BAFTA        | Best Editing        |
|       13 | Cannes       | Palm d’Or           |
|       14 | Oscar        | Best Original Song  |
|       15 | Golden Globe | Best Picture        |
|       16 | Golden Globe | Best Animated Film  |
|       17 | Oscar        | Best Animated Film  |
+----------+--------------+---------------------+
--5. Display all Directors information from the database.

select*from Directors;
```



Output:

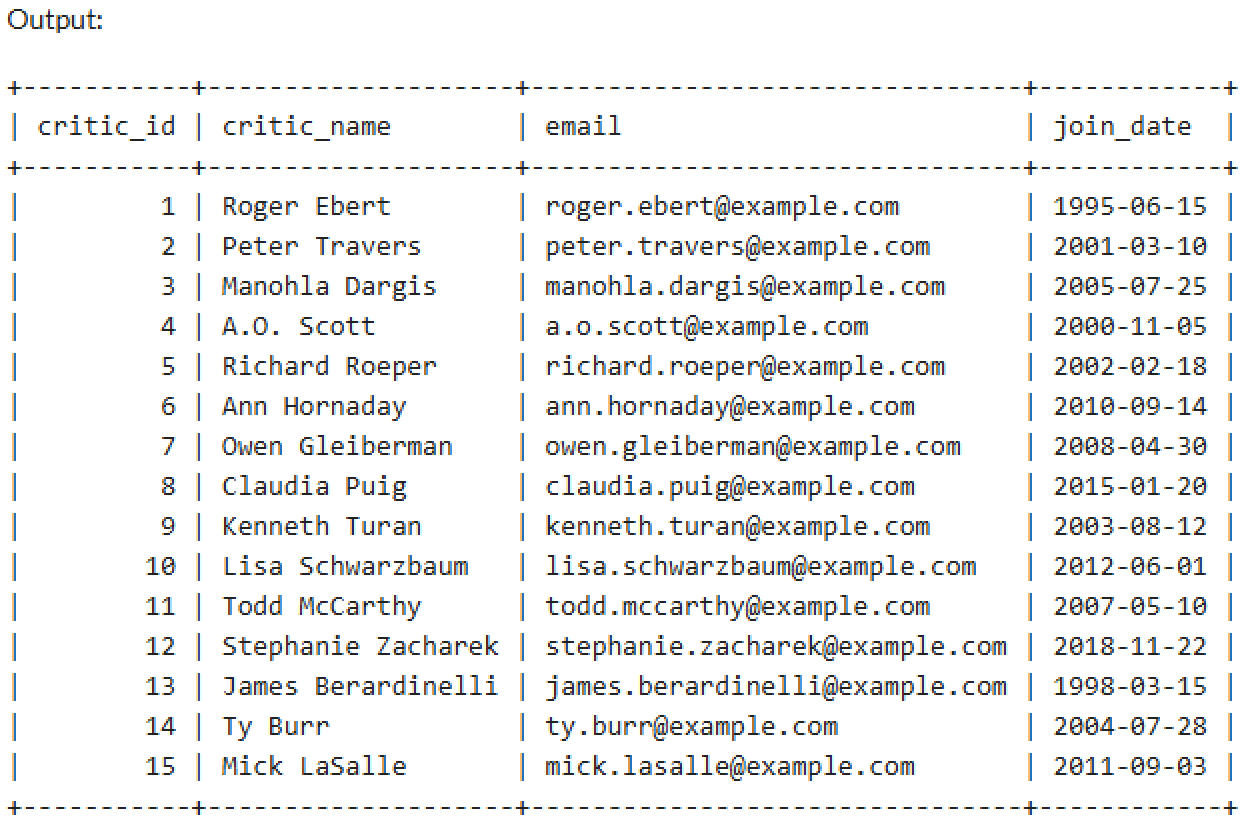
```
+-------------+--------------------+------------+---------------+
| director_id | name               | birth_date | nationality   |
+-------------+--------------------+------------+---------------+
|           1 | Christopher Nolan  | 1970-07-30 | British       |
|           2 | Steven Spielberg   | 1946-12-18 | American      |
|           3 | Martin Scorsese    | 1942-11-17 | American      |
|           4 | Quentin Tarantino  | 1963-03-27 | American      |
|           5 | James Cameron      | 1954-08-16 | Canadian      |
|           6 | Damien Chazelle    | 1983-08-04 | American      |
|           7 | Alfonso Cuarón     | 1961-11-28 | Mexican       |
|           8 | Denis Villeneuve   | 1967-10-03 | Canadian      |
|           9 | Patty Jenkins      | 1971-07-24 | American      |
|          10 | Taika Waititi      | 1975-08-16 | New Zealander |
|          11 | Wes Anderson       | 1969-05-01 | American      |
|          12 | Kathryn Bigelow    | 1951-11-27 | American      |
|          13 | Danny Boyle        | 1956-10-20 | British       |
|          14 | Bong Joon-ho       | 1969-09-14 | South Korean  |
|          15 | Guillermo del Toro | 1964-10-09 | Mexican       |
|          16 | Hayao Miyazaki     | 1941-01-05 | Japanese      |
|          17 | Makoto Shinkai     | 1973-02-09 | Japanese      |
+-------------+--------------------+------------+---------------+
--6. Display all Actors information from the
database.

select*from Actors;
```

Output:



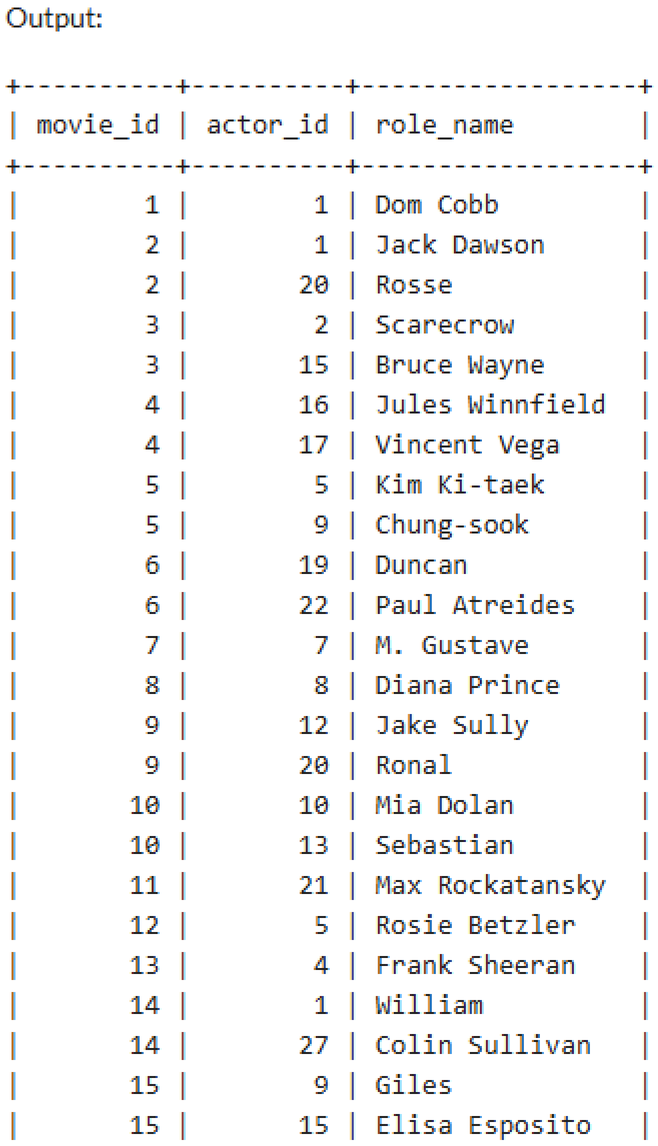
```
+----------+--------------------+------------+-------------+-------------+
| actor_id | name               | birth_date | nationality | total_movie |
+----------+--------------------+------------+-------------+-------------+
|        1 | Leonardo DiCaprio  | 1974-11-11 | American    |          30 |
|        2 | Cillian Murphy     | 1976-05-25 | Irish       |          41 |
|        3 | Tom Hanks          | 1956-07-09 | American    |          40 |
|        4 | Robert De Niro     | 1943-08-17 | American    |          70 |
|        5 | Scarlett Johansson | 1984-11-22 | American    |          35 |
|        6 | Margot Robbie      | 1990-07-02 | Australian  |          30 |
|        7 | Brad Pitt          | 1963-12-18 | American    |          45 |
|        8 | Gal Gadot          | 1990-07-02 | Israeli     |          25 |
|        9 | Denzel Washington  | 1954-12-28 | American    |          50 |
|       10 | Emma Stone         | 1988-11-06 | American    |          20 |
|       11 | Chris Hemsworth    | 1983-08-11 | Australian  |          25 |
|       12 | Sam Worthington    | 1976-08-02 | British     |          35 |
|       13 | Ryan Gosling       | 1980-11-12 | Canadian    |          30 |
|       14 | Anne Hathaway      | 1982-11-12 | American    |          25 |
|       15 | Christian Bale     | 1974-01-30 | British     |          40 |
|       16 | Samuel L. Jackson  | 1948-12-21 | American    |          55 |
|       17 | Robert Downey Jr   | 1965-04-04 | American    |          46 |
|       18 | Dev Patel          | 1984-02-18 | American    |          25 |
|       19 | Jason Momo         | 1979-02-18 | American    |          30 |
|       20 | Kate Winslet       | 1975-10-05 | British     |          30 |
|       21 | Tom Hardy          | 1977-09-15 | British     |          40 |
|       22 | Timothée Chalamet  | 1995-12-27 | American    |          23 |
|       23 | Matthew McConaughey| 1969-11-04 | American    |          36 |
|       24 | George Clooney     | 1961-05-06 | American    |          50 |
|       25 | Jesse Eisenberg    | 1983-10-05 | American    |          30 |
|       26 | Andrew Garfield    | 1983-10-20 | American    |          28 |
|       27 | Matt Damon         | 1970-10-08 | American    |          40 |
|       28 | Keanu Reeves       | 1964-09-02 | Canadian    |          38 |
|       29 | Jeremy Renner      | 1971-01-07 | American    |          43 |
|       30 | Matt Damon         | 1970-10-08 | American    |          40 |
+----------+--------------------+------------+-------------+-------------+
--7. Display all Critics information from the database.
```

select*from Critics;



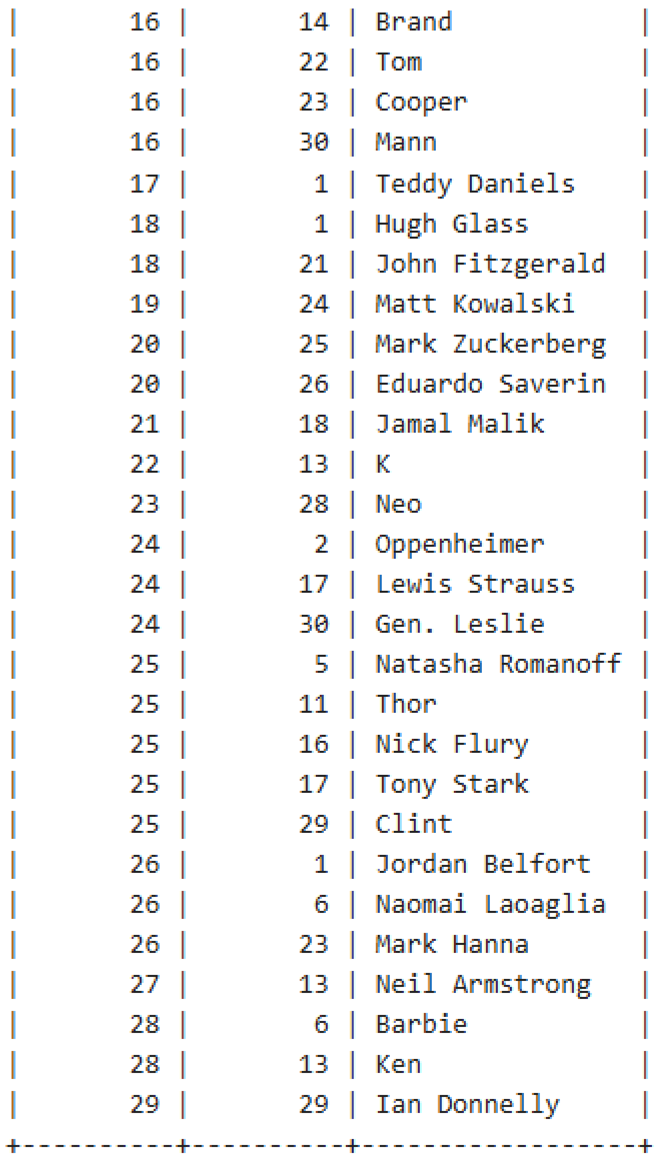
Output:

```
+-----------+--------------------+--------------------------------+------------+
| critic_id | critic_name        | email                          | join_date  |
+-----------+--------------------+--------------------------------+------------+
|         1 | Roger Ebert        | roger.ebert@example.com        | 1995-06-15 |
|         2 | Peter Travers      | peter.travers@example.com      | 2001-03-10 |
|         3 | Manohla Dargis     | manohla.dargis@example.com     | 2005-07-25 |
|         4 | A.O. Scott         | a.o.scott@example.com          | 2000-11-05 |
|         5 | Richard Roeper     | richard.roeper@example.com     | 2002-02-18 |
|         6 | Ann Hornaday       | ann.hornaday@example.com       | 2010-09-14 |
|         7 | Owen Gleiberman    | owen.gleiberman@example.com    | 2008-04-30 |
|         8 | Claudia Puig       | claudia.puig@example.com       | 2015-01-20 |
|         9 | Kenneth Turan      | kenneth.turan@example.com      | 2003-08-12 |
|        10 | Lisa Schwarzbaum   | lisa.schwarzbaum@example.com   | 2012-06-01 |
|        11 | Todd McCarthy      | todd.mccarthy@example.com      | 2007-05-10 |
|        12 | Stephanie Zacharek | stephanie.zacharek@example.com | 2018-11-22 |
|        13 | James Berardinelli | james.berardinelli@example.com | 1998-03-15 |
|        14 | Ty Burr            | ty.burr@example.com            | 2004-07-28 |
|        15 | Mick LaSalle       | mick.lasalle@example.com       | 2011-09-03 |
+-----------+--------------------+--------------------------------+------------+
--8. Display all Movie_Actors information from the database.

select*from Movie_Actors;
```



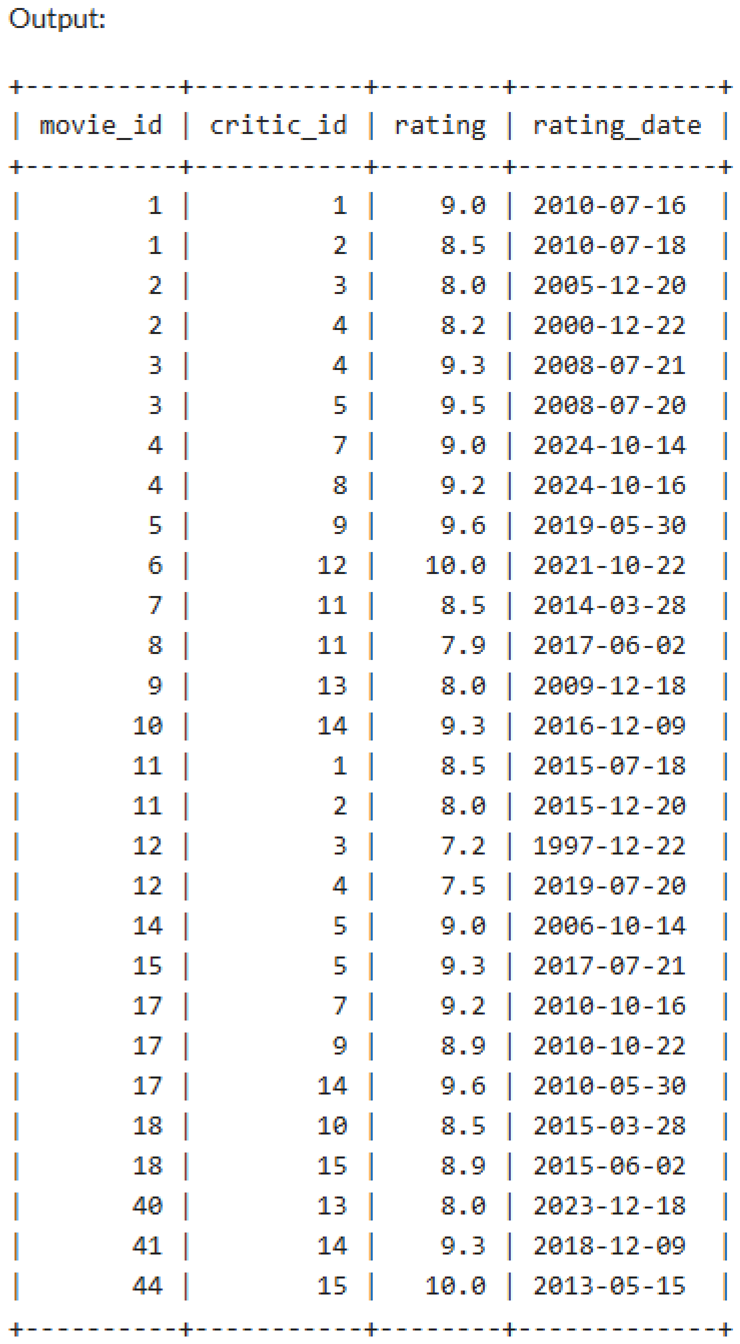
Output:

```
+----------+----------+------------------+
| movie_id | actor_id | role_name        |
+----------+----------+------------------+
|        1 |        1 | Dom Cobb         |
|        2 |        1 | Jack Dawson      |
|        2 |       20 | Rosse            |
|        3 |        2 | Scarecrow        |
|        3 |       15 | Bruce Wayne      |
|        4 |       16 | Jules Winnfield  |
|        4 |       17 | Vincent Vega     |
|        5 |        5 | Kim Ki-taek      |
|        5 |        9 | Chung-sook       |
|        6 |       19 | Duncan           |
|        6 |       22 | Paul Atreides    |
|        7 |        7 | M. Gustave       |
|        8 |        8 | Diana Prince     |
|        9 |       12 | Jake Sully       |
|        9 |       20 | Ronal            |
|       10 |       10 | Mia Dolan        |
|       10 |       13 | Sebastian        |
|       11 |       21 | Max Rockatansky  |
|       12 |        5 | Rosie Betzler    |
|       13 |        4 | Frank Sheeran    |
|       14 |        1 | William          |
|       14 |       27 | Colin Sullivan   |
|       15 |        9 | Giles            |
|       15 |       15 | Elisa Esposito   |
```



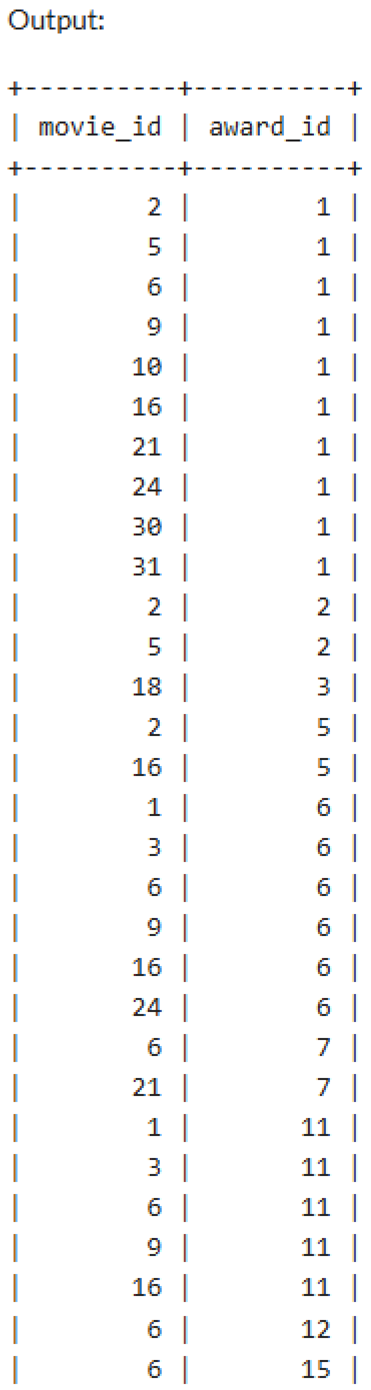
```
|       16 |       14 | Brand            |
|       16 |       22 | Tom              |
|       16 |       23 | Cooper           |
|       16 |       30 | Mann             |
|       17 |        1 | Teddy Daniels    |
|       18 |        1 | Hugh Glass       |
|       18 |       21 | John Fitzgerald  |
|       19 |       24 | Matt Kowalski    |
|       20 |       25 | Mark Zuckerberg  |
|       20 |       26 | Eduardo Saverin  |
|       21 |       18 | Jamal Malik      |
|       22 |       13 | K                |
|       23 |       28 | Neo              |
|       24 |        2 | Oppenheimer      |
|       24 |       17 | Lewis Strauss    |
|       24 |       30 | Gen. Leslie      |
|       25 |        5 | Natasha Romanoff |
|       25 |       11 | Thor             |
|       25 |       16 | Nick Flury       |
|       25 |       17 | Tony Stark       |
|       25 |       29 | Clint            |
|       26 |        1 | Jordan Belfort   |
|       26 |        6 | Naomai Laoaglia  |
|       26 |       23 | Mark Hanna       |
|       27 |       13 | Neil Armstrong   |
|       28 |        6 | Barbie           |
|       28 |       13 | Ken              |
|       29 |       29 | Ian Donnelly     |
+----------+----------+------------------+
--9. Display all Ratings information of movies
from the database.
```

select*from Ratings;



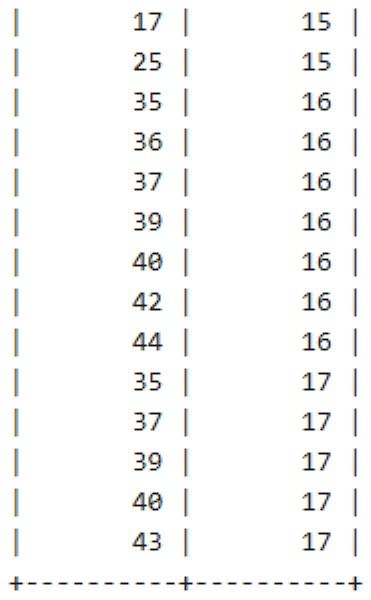
```
Output:

+----------+-----------+--------+-------------+
| movie_id | critic_id | rating | rating_date |
+----------+-----------+--------+-------------+
|        1 |         1 |    9.0 | 2010-07-16  |
|        1 |         2 |    8.5 | 2010-07-18  |
|        2 |         3 |    8.0 | 2005-12-20  |
|        2 |         4 |    8.2 | 2000-12-22  |
|        3 |         4 |    9.3 | 2008-07-21  |
|        3 |         5 |    9.5 | 2008-07-20  |
|        4 |         7 |    9.0 | 2024-10-14  |
|        4 |         8 |    9.2 | 2024-10-16  |
|        5 |         9 |    9.6 | 2019-05-30  |
|        6 |        12 |   10.0 | 2021-10-22  |
|        7 |        11 |    8.5 | 2014-03-28  |
|        8 |        11 |    7.9 | 2017-06-02  |
|        9 |        13 |    8.0 | 2009-12-18  |
|       10 |        14 |    9.3 | 2016-12-09  |
|       11 |         1 |    8.5 | 2015-07-18  |
|       11 |         2 |    8.0 | 2015-12-20  |
|       12 |         3 |    7.2 | 1997-12-22  |
|       12 |         4 |    7.5 | 2019-07-20  |
|       14 |         5 |    9.0 | 2006-10-14  |
|       15 |         5 |    9.3 | 2017-07-21  |
|       17 |         7 |    9.2 | 2010-10-16  |
|       17 |         9 |    8.9 | 2010-10-22  |
|       17 |        14 |    9.6 | 2010-05-30  |
|       18 |        10 |    8.5 | 2015-03-28  |
|       18 |        15 |    8.9 | 2015-06-02  |
|       40 |        13 |    8.0 | 2023-12-18  |
|       41 |        14 |    9.3 | 2018-12-09  |
|       44 |        15 |   10.0 | 2013-05-15  |
+----------+-----------+--------+-------------+
--10. Display all Movie_Awards information of
movies from the database.
```

```
select*from Movie_Awards;
```



Output:

```
+----------+----------+
| movie_id | award_id |
+----------+----------+
|        2 |        1 |
|        5 |        1 |
|        6 |        1 |
|        9 |        1 |
|       10 |        1 |
|       16 |        1 |
|       21 |        1 |
|       24 |        1 |
|       30 |        1 |
|       31 |        1 |
|        2 |        2 |
|        5 |        2 |
|       18 |        3 |
|        2 |        5 |
|       16 |        5 |
|        1 |        6 |
|        3 |        6 |
|        6 |        6 |
|        9 |        6 |
|       16 |        6 |
|       24 |        6 |
|        6 |        7 |
|       21 |        7 |
|        1 |       11 |
|        3 |       11 |
|        6 |       11 |
|        9 |       11 |
|       16 |       11 |
|        6 |       12 |
|        6 |       15 |
```

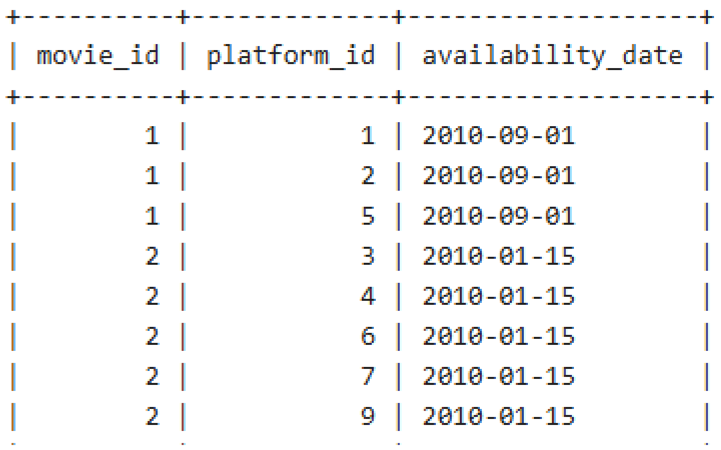


```
|       17 |       15 |
|       25 |       15 |
|       35 |       16 |
|       36 |       16 |
|       37 |       16 |
|       39 |       16 |
|       40 |       16 |
|       42 |       16 |
|       44 |       16 |
|       35 |       17 |
|       37 |       17 |
|       39 |       17 |
|       40 |       17 |
|       43 |       17 |
+----------+----------+
```

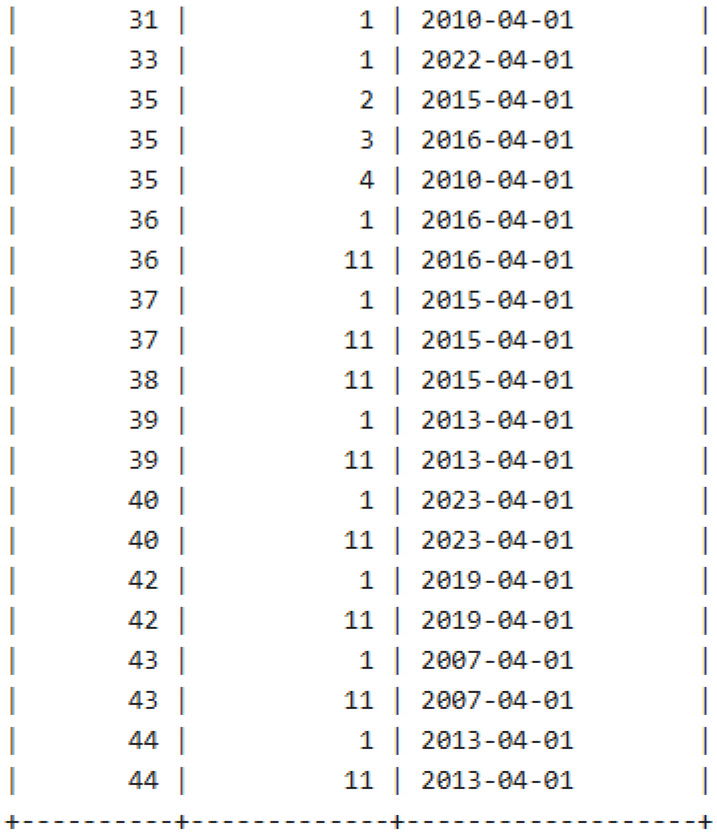
--11. Display all Movie_Platforms information from the database.

select*from Movie_Platforms;

Output:



```
+----------+-------------+-------------------+
| movie_id | platform_id | availability_date |
+----------+-------------+-------------------+
|        1 |           1 | 2010-09-01        |
|        1 |           2 | 2010-09-01        |
|        1 |           5 | 2010-09-01        |
|        2 |           3 | 2010-01-15        |
|        2 |           4 | 2010-01-15        |
|        2 |           6 | 2010-01-15        |
|        2 |           7 | 2010-01-15        |
|        2 |           9 | 2010-01-15        |
|        4 |           4 | 1995-11-01        |
|        5 |           5 | 2020-01-15        |
|        5 |           6 | 2020-01-15        |
|        6 |           1 | 2021-12-01        |
|        6 |           2 | 2021-12-01        |
|        6 |           5 | 2021-12-01        |
|        7 |           7 | 2014-05-01        |
|        8 |           2 | 2017-07-01        |
|        8 |           8 | 2017-07-01        |
|        9 |           1 | 2010-02-15        |
|        9 |           3 | 2010-02-15        |
|        9 |           6 | 2010-02-15        |
|       10 |           1 | 2016-08-15        |
|       10 |           4 | 2016-08-15        |
|       10 |           6 | 2016-08-15        |
|       11 |           1 | 2015-06-01        |
|       12 |          12 | 2020-09-15        |
|       13 |          13 | 2019-12-01        |
|       14 |          14 | 2018-07-15        |
|       15 |          14 | 2021-04-01        |
|       16 |           1 | 2014-04-01        |
|       16 |           5 | 2014-04-01        |
|       16 |           6 | 2014-04-01        |
|       17 |           1 | 2010-04-01        |
|       17 |           5 | 2010-04-01        |
|       17 |           8 | 2010-04-01        |
|       18 |           1 | 2015-04-01        |
|       20 |           1 | 2010-04-01        |
|       21 |           1 | 2008-04-01        |
|       22 |           1 | 2017-04-01        |
|       23 |           1 | 2010-04-01        |
|       24 |           1 | 2023-04-01        |
|       25 |           1 | 2012-04-01        |
|       26 |           1 | 2013-04-01        |
|       28 |           1 | 2023-04-01        |
|       30 |           1 | 2010-04-01        |
```



```
|       31 |           1 | 2010-04-01        |
|       33 |           1 | 2022-04-01        |
|       35 |           2 | 2015-04-01        |
|       35 |           3 | 2016-04-01        |
|       35 |           4 | 2010-04-01        |
|       36 |           1 | 2016-04-01        |
|       36 |          11 | 2016-04-01        |
|       37 |           1 | 2015-04-01        |
|       37 |          11 | 2015-04-01        |
|       38 |          11 | 2015-04-01        |
|       39 |           1 | 2013-04-01        |
|       39 |          11 | 2013-04-01        |
|       40 |           1 | 2023-04-01        |
|       40 |          11 | 2023-04-01        |
|       42 |           1 | 2019-04-01        |
|       42 |          11 | 2019-04-01        |
|       43 |           1 | 2007-04-01        |
|       43 |          11 | 2007-04-01        |
|       44 |           1 | 2013-04-01        |
|       44 |          11 | 2013-04-01        |
+----------+-------------+-------------------+
```

(Advanced Queries)

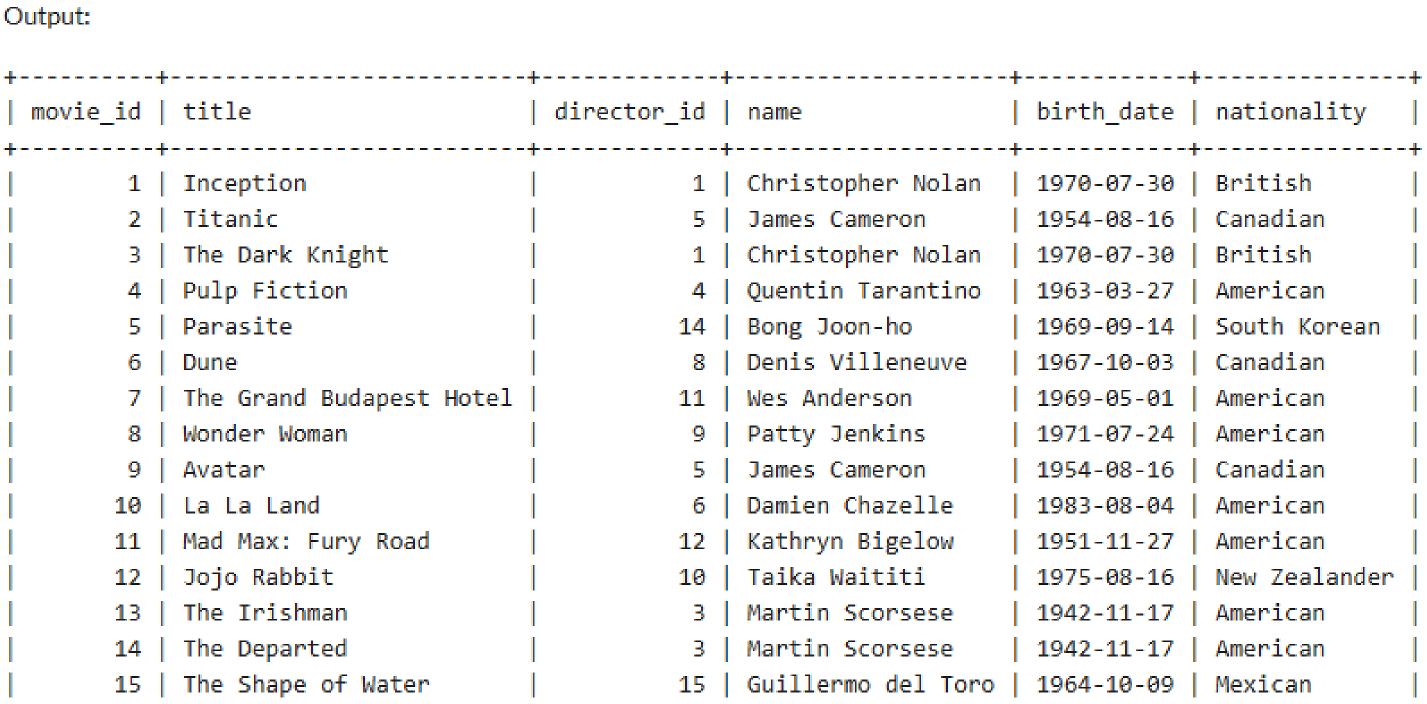
--12. Display all movies directed by their director.

--Natural Join—

select movie_id,title,Directors.director_id,name,birth_date,nationality

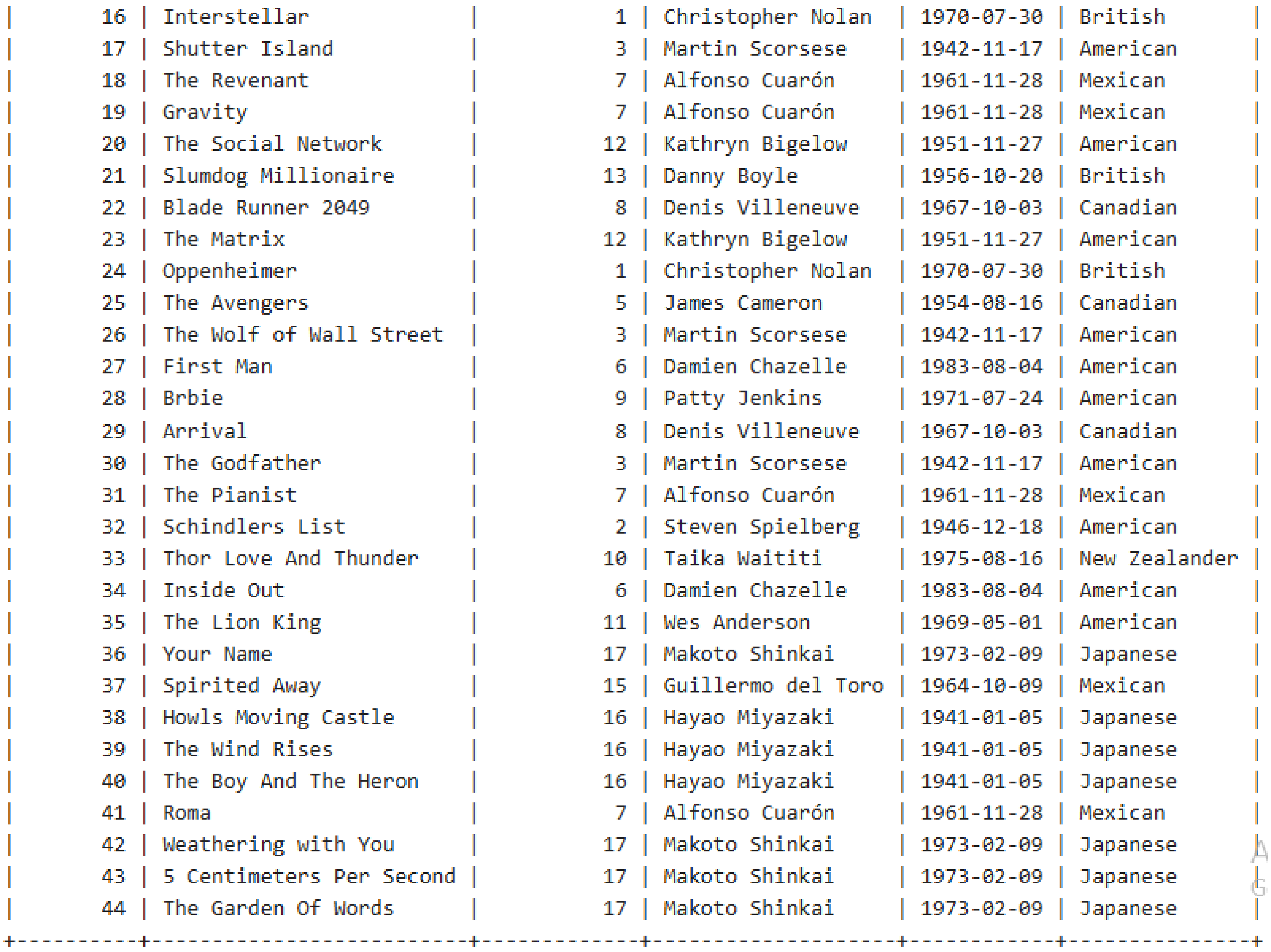
from Movies **natural join** Directors

order by movie_id;



Output:

```
+----------+--------------------------+-------------+--------------------+------------+---------------+
| movie_id | title                    | director_id | name               | birth_date | nationality   |
+----------+--------------------------+-------------+--------------------+------------+---------------+
|        1 | Inception                |           1 | Christopher Nolan  | 1970-07-30 | British       |
|        2 | Titanic                  |           5 | James Cameron      | 1954-08-16 | Canadian      |
|        3 | The Dark Knight          |           1 | Christopher Nolan  | 1970-07-30 | British       |
|        4 | Pulp Fiction             |           4 | Quentin Tarantino  | 1963-03-27 | American      |
|        5 | Parasite                 |          14 | Bong Joon-ho       | 1969-09-14 | South Korean  |
|        6 | Dune                     |           8 | Denis Villeneuve   | 1967-10-03 | Canadian      |
|        7 | The Grand Budapest Hotel |          11 | Wes Anderson       | 1969-05-01 | American      |
|        8 | Wonder Woman             |           9 | Patty Jenkins      | 1971-07-24 | American      |
|        9 | Avatar                   |           5 | James Cameron      | 1954-08-16 | Canadian      |
|       10 | La La Land               |           6 | Damien Chazelle    | 1983-08-04 | American      |
|       11 | Mad Max: Fury Road       |          12 | Kathryn Bigelow    | 1951-11-27 | American      |
|       12 | Jojo Rabbit              |          10 | Taika Waititi      | 1975-08-16 | New Zealander |
|       13 | The Irishman             |           3 | Martin Scorsese    | 1942-11-17 | American      |
|       14 | The Departed             |           3 | Martin Scorsese    | 1942-11-17 | American      |
|       15 | The Shape of Water       |          15 | Guillermo del Toro | 1964-10-09 | Mexican       |
```



```
|       16 | Interstellar             |           1 | Christopher Nolan  | 1970-07-30 | British       |
|       17 | Shutter Island           |           3 | Martin Scorsese    | 1942-11-17 | American      |
|       18 | The Revenant             |           7 | Alfonso Cuarón     | 1961-11-28 | Mexican       |
|       19 | Gravity                  |           7 | Alfonso Cuarón     | 1961-11-28 | Mexican       |
|       20 | The Social Network       |          12 | Kathryn Bigelow    | 1951-11-27 | American      |
|       21 | Slumdog Millionaire      |          13 | Danny Boyle        | 1956-10-20 | British       |
|       22 | Blade Runner 2049        |           8 | Denis Villeneuve   | 1967-10-03 | Canadian      |
|       23 | The Matrix               |          12 | Kathryn Bigelow    | 1951-11-27 | American      |
|       24 | Oppenheimer              |           1 | Christopher Nolan  | 1970-07-30 | British       |
|       25 | The Avengers             |           5 | James Cameron      | 1954-08-16 | Canadian      |
|       26 | The Wolf of Wall Street  |           3 | Martin Scorsese    | 1942-11-17 | American      |
|       27 | First Man                |           6 | Damien Chazelle    | 1983-08-04 | American      |
|       28 | Brbie                    |           9 | Patty Jenkins      | 1971-07-24 | American      |
|       29 | Arrival                  |           8 | Denis Villeneuve   | 1967-10-03 | Canadian      |
|       30 | The Godfather            |           3 | Martin Scorsese    | 1942-11-17 | American      |
|       31 | The Pianist              |           7 | Alfonso Cuarón     | 1961-11-28 | Mexican       |
|       32 | Schindlers List          |           2 | Steven Spielberg   | 1946-12-18 | American      |
|       33 | Thor Love And Thunder    |          10 | Taika Waititi      | 1975-08-16 | New Zealander |
|       34 | Inside Out               |           6 | Damien Chazelle    | 1983-08-04 | American      |
|       35 | The Lion King            |          11 | Wes Anderson       | 1969-05-01 | American      |
|       36 | Your Name                |          17 | Makoto Shinkai     | 1973-02-09 | Japanese      |
|       37 | Spirited Away            |          15 | Guillermo del Toro | 1964-10-09 | Mexican       |
|       38 | Howls Moving Castle      |          16 | Hayao Miyazaki     | 1941-01-05 | Japanese      |
|       39 | The Wind Rises           |          16 | Hayao Miyazaki     | 1941-01-05 | Japanese      |
|       40 | The Boy And The Heron    |          16 | Hayao Miyazaki     | 1941-01-05 | Japanese      |
|       41 | Roma                     |           7 | Alfonso Cuarón     | 1961-11-28 | Mexican       |
|       42 | Weathering with You      |          17 | Makoto Shinkai     | 1973-02-09 | Japanese      |
|       43 | 5 Centimeters Per Second |          17 | Makoto Shinkai     | 1973-02-09 | Japanese      |
|       44 | The Garden Of Words      |          17 | Makoto Shinkai     | 1973-02-09 | Japanese      |
+----------+--------------------------+-------------+--------------------+------------+---------------+
--13. Display all movies who won awards.
```

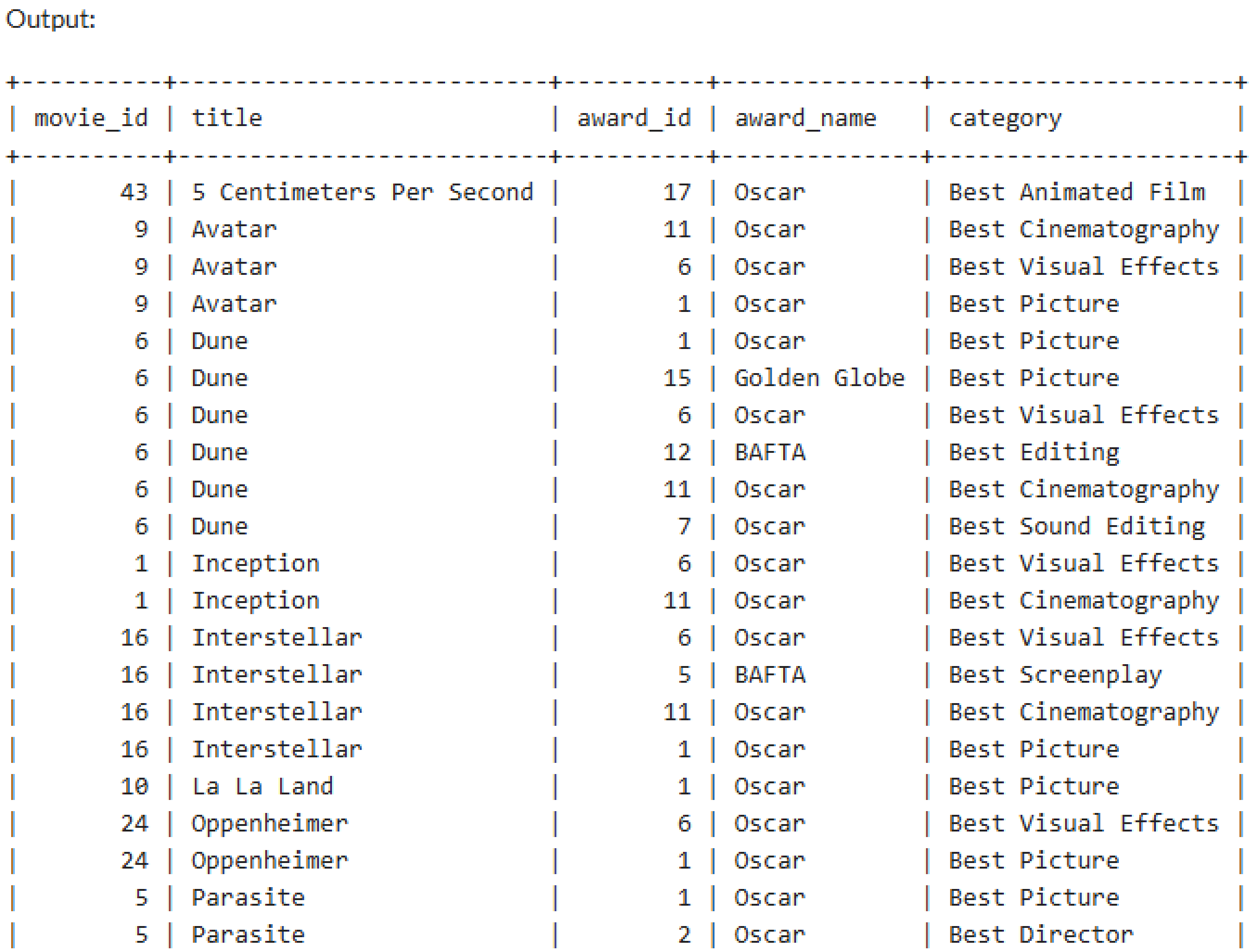
--join operation—

select Movies.movie_id,Movies.title,Awards.award_id,award_name,category

from Movies **join** Movie_Awards **join** Awards

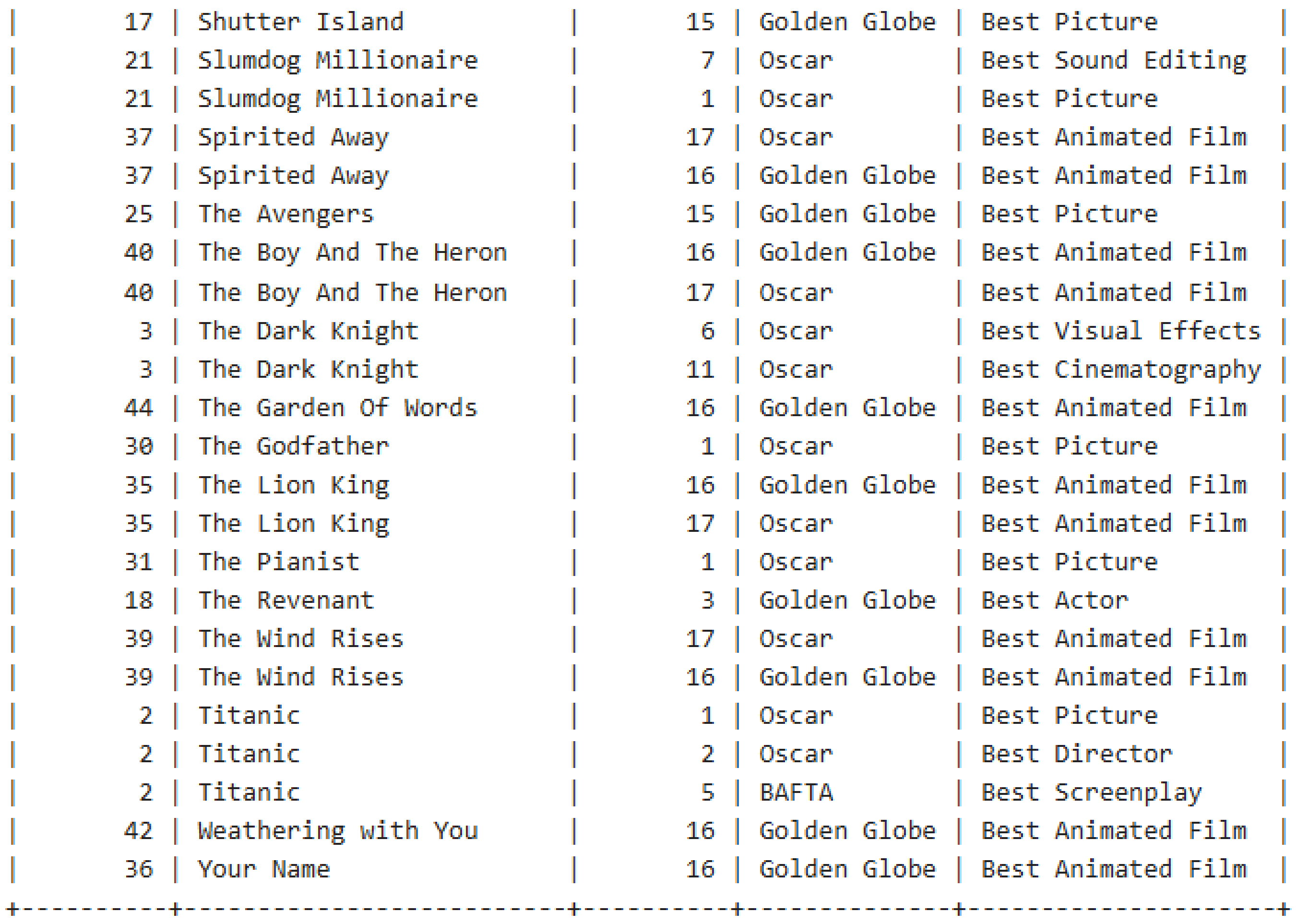
on Movies.movie_id=Movie_Awards.movie_id

and Movie_Awards.award_id=Awards.award_id;



Output:

```
+----------+-------------------------+----------+-------------+---------------------+
| movie_id | title                   | award_id | award_name  | category            |
+----------+-------------------------+----------+-------------+---------------------+
|       43 | 5 Centimeters Per Second |       17 | Oscar       | Best Animated Film  |
|        9 | Avatar                  |       11 | Oscar       | Best Cinematography |
|        9 | Avatar                  |        6 | Oscar       | Best Visual Effects |
|        9 | Avatar                  |        1 | Oscar       | Best Picture        |
|        6 | Dune                    |        1 | Oscar       | Best Picture        |
|        6 | Dune                    |       15 | Golden Globe | Best Picture       |
|        6 | Dune                    |        6 | Oscar       | Best Visual Effects |
|        6 | Dune                    |       12 | BAFTA       | Best Editing        |
|        6 | Dune                    |       11 | Oscar       | Best Cinematography |
|        6 | Dune                    |        7 | Oscar       | Best Sound Editing  |
|        1 | Inception               |        6 | Oscar       | Best Visual Effects |
|        1 | Inception               |       11 | Oscar       | Best Cinematography |
|       16 | Interstellar            |        6 | Oscar       | Best Visual Effects |
|       16 | Interstellar            |        5 | BAFTA       | Best Screenplay     |
|       16 | Interstellar            |       11 | Oscar       | Best Cinematography |
|       16 | Interstellar            |        1 | Oscar       | Best Picture        |
|       10 | La La Land              |        1 | Oscar       | Best Picture        |
|       24 | Oppenheimer             |        6 | Oscar       | Best Visual Effects |
|       24 | Oppenheimer             |        1 | Oscar       | Best Picture        |
|        5 | Parasite                |        1 | Oscar       | Best Picture        |
|        5 | Parasite                |        2 | Oscar       | Best Director       |
```

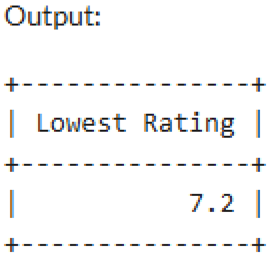


```
|       17 | Shutter Island            |       15 | Golden Globe | Best Picture        |
|       21 | Slumdog Millionaire       |        7 | Oscar        | Best Sound Editing  |
|       21 | Slumdog Millionaire       |        1 | Oscar        | Best Picture        |
|       37 | Spirited Away             |       17 | Oscar        | Best Animated Film  |
|       37 | Spirited Away             |       16 | Golden Globe | Best Animated Film  |
|       25 | The Avengers              |       15 | Golden Globe | Best Picture        |
|       40 | The Boy And The Heron     |       16 | Golden Globe | Best Animated Film  |
|       40 | The Boy And The Heron     |       17 | Oscar        | Best Animated Film  |
|        3 | The Dark Knight           |        6 | Oscar        | Best Visual Effects |
|        3 | The Dark Knight           |       11 | Oscar        | Best Cinematography |
|       44 | The Garden Of Words       |       16 | Golden Globe | Best Animated Film  |
|       30 | The Godfather             |        1 | Oscar        | Best Picture        |
|       35 | The Lion King             |       16 | Golden Globe | Best Animated Film  |
|       35 | The Lion King             |       17 | Oscar        | Best Animated Film  |
|       31 | The Pianist               |        1 | Oscar        | Best Picture        |
|       18 | The Revenant              |        3 | Golden Globe | Best Actor          |
|       39 | The Wind Rises            |       17 | Oscar        | Best Animated Film  |
|       39 | The Wind Rises            |       16 | Golden Globe | Best Animated Film  |
|        2 | Titanic                   |        1 | Oscar        | Best Picture        |
|        2 | Titanic                   |        2 | Oscar        | Best Director       |
|        2 | Titanic                   |        5 | BAFTA        | Best Screenplay     |
|       42 | Weathering with You       |       16 | Golden Globe | Best Animated Film  |
|       36 | Your Name                 |       16 | Golden Globe | Best Animated Film  |
+----------+---------------------------+----------+--------------+---------------------+
```

--14. Display the lowest rating movie from the database.

--Min / max / avg / count / sum--

```
select min(rating) as 'Lowest Rating'

from movies,critics,Ratings

where Movies.movie_id=Ratings.movie_id

and Ratings.critic_id=critics.critic_id;
```



Output:

```
+----------------+
| Lowest Rating |
+----------------+
|            7.2 |
+----------------+
```

--15. Display movies based on genre which is grater then 4

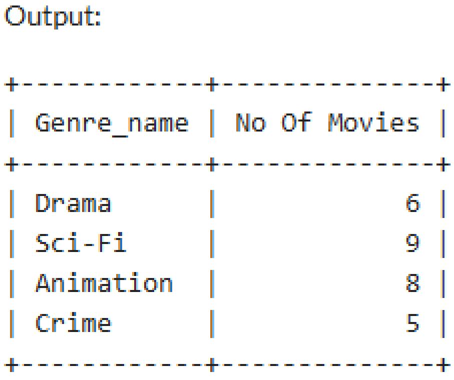
--Aggregate Functions(using group by & having)--

select Genre_name,count(movie_id) as 'No Of Movies'

from Movies **natural join** Genres

group by Genre_name

having count(movie_id)>4;



Output:

```
+------------+--------------+
| Genre_name | No Of Movies |
+------------+--------------+
| Drama      |            6 |
| Sci-Fi     |            9 |
| Animation  |            8 |
| Crime      |            5 |
+------------+--------------+
```

--16. Show the total number of films available by industry.

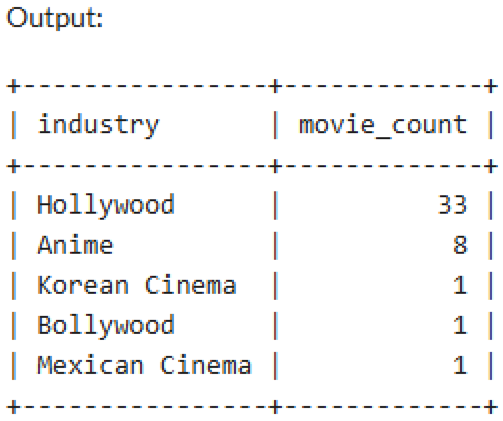
-- GROUP BY & ORDER BY—

SELECT industry, COUNT(movie_id) AS movie_count

FROM Movies

GROUP BY industry

ORDER BY movie_count DESC;



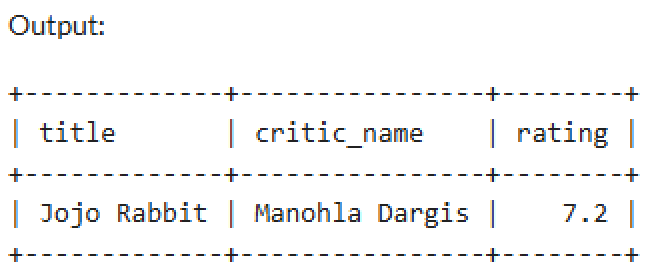
Output:

```
+----------------+-------------+
| industry       | movie_count |
+----------------+-------------+
| Hollywood      |          33 |
| Anime          |           8 |
| Korean Cinema  |           1 |
| Bollywood      |           1 |
| Mexican Cinema |           1 |
+----------------+-------------+
```

--17. Display critic name who gave lowest rate to a movie from the database.

--Subqueries—

```
select title,critic_name,rating
from Movies m join critics c join Ratings r
on c.critic_id=r.critic_id
and m.movie_id=r.movie_id
and rating in
(select min(rating)
from Movies m join Ratings r
on m.movie_id=r.movie_id);
```



Output:

```
+-------------+---------------+--------+
| title       | critic_name   | rating |
+-------------+---------------+--------+
| Jojo Rabbit | Manohla Dargis |    7.2 |
+-------------+---------------+--------+
```

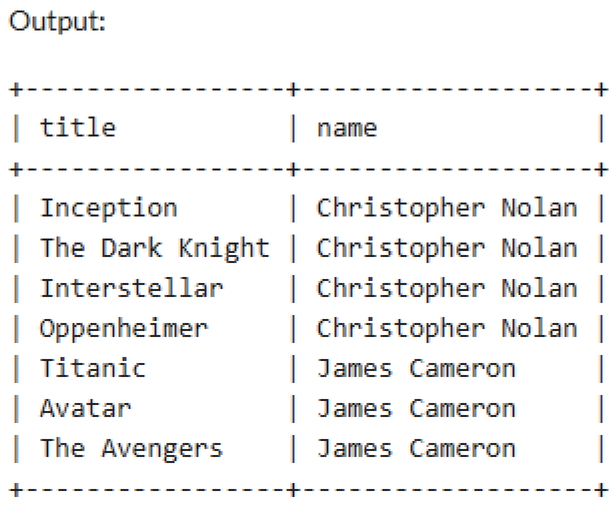
--18. Display movie list which is directed by Christopher Nolan or James Cameron .

--Set Operations --

(select title,name from movies **natural join** Directors where name='Christopher Nolan')

union

(select title,name from movies **natural join** Directors where name='James Cameron');



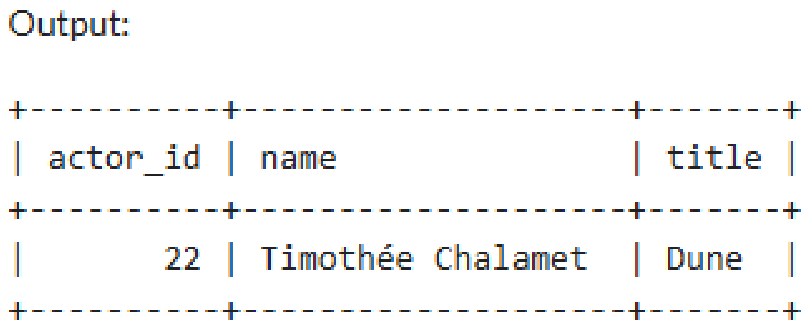
Output:

```
+-----------------+-------------------+
| title           | name              |
+-----------------+-------------------+
| Inception       | Christopher Nolan |
| The Dark Knight | Christopher Nolan |
| Interstellar    | Christopher Nolan |
| Oppenheimer     | Christopher Nolan |
| Titanic         | James Cameron     |
| Avatar          | James Cameron     |
| The Avengers    | James Cameron     |
+-----------------+-------------------+
```

--19. Display actors list who’s name start with ‘T’ and who acted in a movie that title start with ‘D’ .

--String Operation --

```
select  a.actor_id,name,title
from Actors as a, Movie_Actors as ma,Movies as m
where a.actor_id=ma.actor_id
and ma.movie_id=m.movie_id
and a.name like 'T%' and title like 'D%';
```



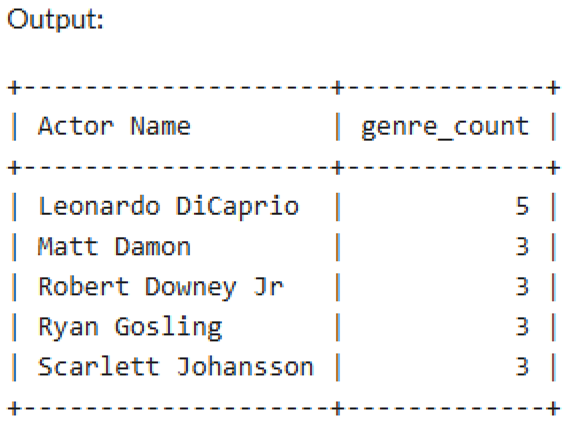
Output:

```
+----------+--------------------+-------+
| actor_id | name               | title |
+----------+--------------------+-------+
|       22 | Timothée Chalamet  | Dune  |
+----------+--------------------+-------+
```

(Complex Queries)

--20.Actors with roles across different genres.

```
SELECT a.name AS 'Actor Name', COUNT(DISTINCT g.genre_id) AS genre_count

FROM Actors a JOIN Movie_Actors ma JOIN Movies m JOIN Genres g

ON a.actor_id = ma.actor_id

and ma.movie_id = m.movie_id

and m.genre_id = g.genre_id

GROUP BY a.name

having genre_count>2

ORDER BY genre_count DESC;
```

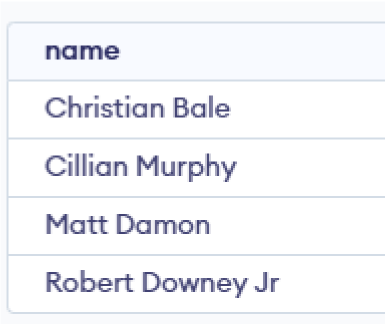


Output:

```
+--------------------+-------------+
| Actor Name         | genre_count |
+--------------------+-------------+
| Leonardo DiCaprio  |           5 |
| Matt Damon         |           3 |
| Robert Downey Jr   |           3 |
| Ryan Gosling       |           3 |
| Scarlett Johansson |           3 |
+--------------------+-------------+
```

--21.Find actors who have acted in movies released after 2015 and are also present in movies released before 2010.

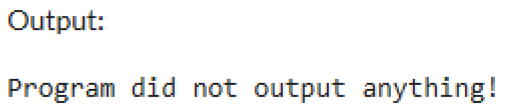
```
SELECT name
FROM Actors
WHERE actor_id IN (
SELECT actor_id
FROM Movie_Actors
WHERE movie_id IN (
SELECT movie_id
FROM Movies
WHERE release_year > 2015))
INTERSECT
SELECT name
FROM Actors
WHERE actor_id IN (
SELECT actor_id
FROM Movie_Actors
WHERE movie_id IN (
SELECT movie_id
FROM Movies
WHERE release_year < 2010));
```



name
Christian Bale
Cillian Murphy
Matt Damon
Robert Downey Jr

--22.Find actors who act in all type of movie genres.

```
SELECT a.name AS actor_name
FROM Actors a
JOIN Movie_Actors ma ON a.actor_id = ma.actor_id
JOIN Movies m ON ma.movie_id = m.movie_id
JOIN Genres g ON m.genre_id = g.genre_id
GROUP BY a.actor_id, a.name
HAVING COUNT(DISTINCT g.genre_id) = (SELECT COUNT(*) FROM Genres);
```

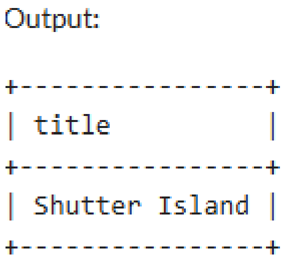


Output:

```
Program did not output anything!
```

--23. Lists movies with average ratings above 9 by more than two critics.

```
SELECT m.title

FROM Movies m

JOIN Ratings r ON m.movie_id = r.movie_id

GROUP BY m.title

HAVING COUNT(r.critic_id) > 2 AND AVG(r.rating) > 9.0;
```

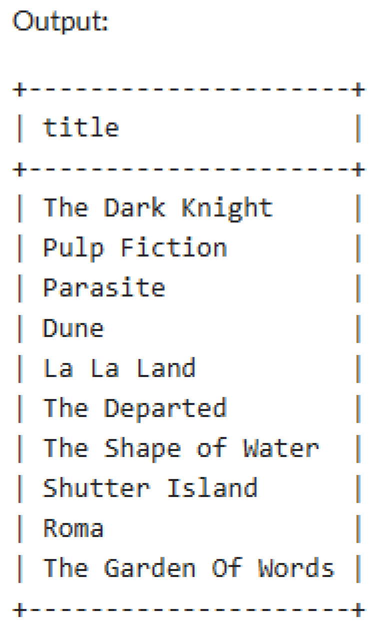


Output:

```
+-----------------+
| title           |
+-----------------+
| Shutter Island |
+-----------------+
```

--24.Find movies that have an average rating higher than the overall average rating.

```
SELECT title
FROM Movies
WHERE movie_id IN (
    SELECT movie_id
    FROM Ratings
    GROUP BY movie_id
    HAVING AVG(rating) > (
        SELECT AVG(rating) FROM Ratings
    )
);
```

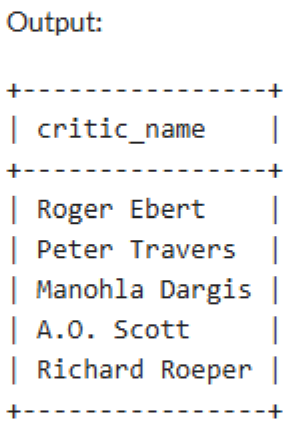


Output:

```
+----------------------+
| title                |
+----------------------+
| The Dark Knight      |
| Pulp Fiction         |
| Parasite             |
| Dune                 |
| La La Land           |
| The Departed         |
| The Shape of Water   |
| Shutter Island       |
| Roma                 |
| The Garden Of Words  |
+----------------------+
```

--25.Top 5 Critics Who Rated the Most Movies .

```
SELECT critic_name

FROM Critics

WHERE critic_id

IN (

   SELECT critic_id

   FROM Ratings

   GROUP BY critic_id

   ORDER BY COUNT(movie_id) DESC


)

limit 5;
```

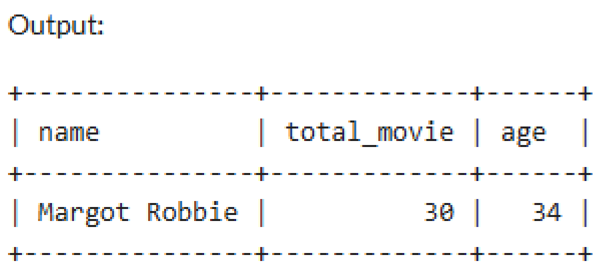


Output:

```
+----------------+
| critic_name    |
+----------------+
| Roger Ebert    |
| Peter Travers  |
| Manohla Dargis |
| A.O. Scott     |
| Richard Roeper |
+----------------+
```

--26.Find the youngest actor who act more then 20 movies and act more then 1 type of genre.

```
SELECT name, total_movie, (2024- YEAR( birth_date)) AS age
FROM Actors
where total_movie > 20
and actor_id in
(
SELECT a.actor_id
FROM Actors a
JOIN Movie_Actors ma ON a.actor_id = ma.actor_id
JOIN Movies m ON ma.movie_id = m.movie_id
JOIN Genres g ON m.genre_id = g.genre_id
GROUP BY a.actor_id, a.name
HAVING COUNT(DISTINCT g.genre_id)>1
)
ORDER BY age ASC
limit 1;
```



Output:

```
+---------------+-------------+------+
| name          | total_movie | age  |
+---------------+-------------+------+
| Margot Robbie |          30 |   34 |
+---------------+-------------+------+
```

❑ Challenges & Solutions

- **Challenges:**
  - Handling large datasets efficiently.
  - Ensuring accurate recommendations.
- **Solutions:**
  - Optimized SQL queries for faster performance.
  - Regular updates to the database for accuracy.

❑ Conclusion

A Movie Recommendation Database Management System concludes with the critical insight that such systems enhance user experiences by personalizing content delivery. Through techniques like based on Genres , Actors information, Directors Information, released in specific Platforms , content-based filtering, or hybrid methods, they analyze user preferences to suggest relevant movies. The final step to optimize the system is "Continuous Feedback Integration". By incorporating user feedback on recommendations, the system refines its algorithms, ensuring higher accuracy and adaptability over time. This iterative improvement not only boosts user satisfaction but also ensures the system stays relevant in a dynamic entertainment landscape.

❑ CEP

How Movie Recommendation Database Management System addresses Complex Engineering Problem (CEP) according to Washington Accord :

Knowledge Profile : K3,K5,K6

Problem Solving : P1,P3,P7

Activities : A1,A5

Mapping the Knowledge Profiles (Ks) to The Movie Recommendation Database Management System Project:

Ks	Attribute	How Ks are addressed through the project
K3	**Engineering Fundamentals**	The **relational database design** to represent entities like Genres, Directors, , Movie_Awards, Movie_Actors. The system employs normalization (1NF, 2NF, 3NF) to avoid data redundancy and maintain integrity, ensuring accurate relationships between tables (e.g. Movies→ Directors, Movies→ Genres).
K5	**Engineering Design**	Relational Database Architecture (RDBMS), Design Decisions **:** Primary and Foreign Keys , ACID (Atomicity, Consistency, Isolation, Durability) Compliance, Support for Complex Queries , Data Scalability , Indexing
K6	**Engineering Practice (Technology)**	SQL for Data Manipulation and Queries , Microsoft SQL Server Management Studio, RDBMS Built-In Transaction Management, ERD (Entity-Relationship Diagram) Tools like Lucidchart or dbdiagram.io

Mapping the Problem-Solving Attributes (Ps) to The Movie Recommendation Database Management System Project:

Ps	Attribute	How Ps are addressed through the project
P1	**Depth of Knowledge Required**	The **Movie Recommendation Database Management System** involves a strong understanding of relational database **design, normalization,** and **SQL query optimization**. The system requires defining relationships between entities like Movies_Awards, Movie_Actors, and Movie_Platforms using **foreign keys** and **primary keys**, along with **Entity- Relationship Diagrams (ERDs). Normalization** ensures data integrity and efficiency by reducing redundancy.
P3	**Depth of Analysis Required**	The work required an in-depth analysis of the database schema to handle complex relationships among entities such as Movie_Actors, Movie_Awards, Directors, Critics. Advanced SQL queries and optimizations were analyzed to efficiently retrieve data for movie recommendation reports.
P7	**Interdependence**	**Sub-problems and Dependencies**: For example, A movie recommendation system predicts user preferences by analyzing data like ratings, watch history, and movie attributes. For example, if a user enjoys The Dark Knight, the system might suggest *Interstellar* by identifying similar user behaviors or shared movie features like genre or director. Continuous feedback refines these suggestions for a more personalized experience. The system must handle these dependencies efficiently, demonstrating high interdependence between different data entities.

Mapping the Activities (As) to The Movie Recommendation Database Management System Project:

As	Attribute	How As are addressed through the project
A1	**Range of Resources Involved**	The project involves a range of resources including Movies (Actors, Directors, Platforms,Awards), **hardware resources** (PC), **software resources** (database management system (Microsoft SQL Server), and **data resources** (Movies, Awards, Actors profile, etc.). Efficient resource management and coordination were essential for maintaining system integrity and security.
A5	**Familiarity**	The project involves handling and managing complex and rarely encountered scenarios in movie recommendation**.** Managing diverse and unstructured data such as Movie_Actrors, Movie_Platforms, and Ratings statements. Designing a database capable of accommodating **dynamic workflows**, such as updates from Critics review or releases in multiple Movie_Platforms. Implementing advanced querying mechanisms to find information related to a Movies or Actors .

Github Link:

https://github.com/Sejonhossain/Database_Final_Project