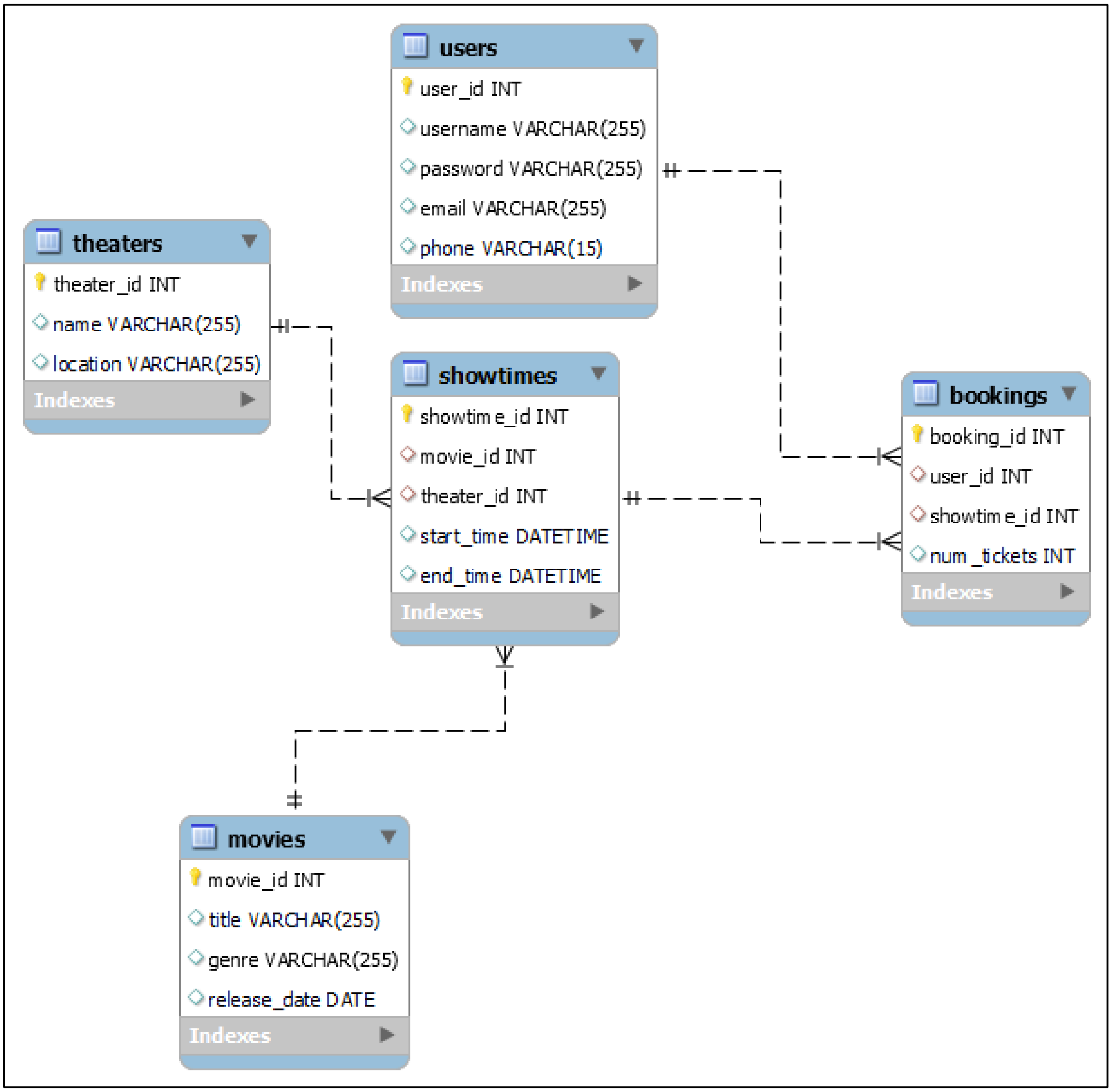
Assignment 1

Problem Statement: - Creating a MySQL project for an Online Movie Ticket Booking System (BookMyShow). BookMyShow-like application involves designing a database to manage information related to movies, theaters, bookings, users, and other relevant entities.

ER-Diagram: -



Database Schema:

1. Users Table: -

user_id (Primary Key), username, password, email, phone

Users				
user_id	**username**	**password**	**email**	**phone**
1	Sanjana	Pass123	Sanjana@email.com	123-456-7890
2	Sumit	SecurePass	Sumit@email.com	987-654-3210
3	Apurva	StrongPassword	Apurva@email.com	555-123-4567
4	Amit	Secret123	Amit@email.com	111-222-3333
5	Tejashri	Password123	Tejashri@email.com	999-888-7777
6	Jayesh	DoePass123	Jayesh@email.com	123-987-6543
7	Hati	MiaSecure	Hati@email.com	555-777-1234
8	Harsh	SamPass789	Harsh@email.com	987-234-5678
9	Manas	SophiePass	Manas@email.com	111-333-9999
10	Seanna	Michael123	Seanna@email.com	777-222-4444
11	Swapnil	MaxPass123	Swapnil@email.com	123-555-7890
12	Priya	BellaSecure	Priya@email.com	999-888-7777
13	Omkar	LucasStrong	Omkar@email.com	555-222-3333
14	Shivam	Ava123Pass	Shivam@email.com	111-444-5555
15	Shashank	SmithPass789	Shashank@email.com	777-666-8888
16	Parag	JacksonPass	Parag@email.com	123-987-6543
17	Shubham	SophiaSecure	Shubham@email.com	555-777-1234
18	Niketan	LoganPass789	Niketan@email.com	987-234-5678
19	Mrudul	AriaPass	Mrudul@email.com	111-333-9999
20	Prachi	Benjamin123	Prachi@email.com	777-222-4444
21	Sarthak	NoahPass123	Sarthak@email.com	123-555-7890
22	Saloni	GraceSecure	Saloni@email.com	999-888-7777
23	Pranav	EthanStrong	Pranav@email.com	555-222-3333
24	Komal	LilyAvaPass	Komal@email.com	111-444-5555
25	Deepak	WilliamPass789	Deepak@email.com	777-666-8888

2. Movies Table: -

movie_id (Primary Key), title, genre, release_date

Movies			
movie_id	**Title**	**genre**	**release_date**
101	The Matrix	Action	3/31/1999
102	Inception	Sci-Fi	7/16/2010
103	The Shawshank Redemption	Drama	9/23/1994

104	Forrest Gump	Drama	7/6/1994
105	The Dark Knight	Action	7/18/2008
106	Pulp Fiction	Crime	10/14/1994
107	Fight Club	Drama	10/15/1999
108	The Godfather	Crime	3/24/1972
109	Titanic	Romance	12/19/1997
110	The Lord of the Rings: The Fellowship of the Ring	Fantasy	12/19/2001
111	Jurassic Park	Adventure	6/11/1993
112	Avatar	Action	12/18/2009
113	The Silence of the Lambs	Thriller	2/14/1991
114	The Social Network	Biography	9/24/2010
115	The Shining	Horror	5/23/1980
116	The Departed	Crime	10/6/2006
117	The Wizard of Oz	Fantasy	8/25/1939
118	The Matrix Reloaded	Action	5/15/2003
119	Eternal Sunshine of the Spotless Mind	Drama	3/19/2004
120	The Lion King	Animation	6/15/1994
121	Gladiator	Action	5/5/2000
122	Inglourious Basterds	War	8/21/2009
123	Back to the Future	Adventure	7/3/1985
124	The Godfather: Part II	Crime	12/20/1974
125	The Avengers	Action	4/11/2012

3. Theaters Table: -

theater_id (Primary Key), name, location

Theaters		
theater_id	**name**	**location**
1	Cinema Paradise	New York
2	City Lights Cinema	Los Angeles
3	Downtown Cinemas	Chicago
4	Metroplex Movies	Houston
5	Sunset Theatres	Phoenix
6	Golden Gate Cinemas	San Francisco
7	Midtown Movies	Philadelphia
8	Lakeside Theatres	Dallas
9	Central Cinema	Miami
10	Capitol Cinemas	Atlanta
11	Hollywood Theaters	Seattle
12	Riverside Cinemas	Denver
13	Liberty Theatres	Boston

14	Harborfront Movies	San Diego
15	Magnolia Cinemas	Minneapolis
16	Grandview Theatres	Portland
17	Downtown Cinemas	Detroit
18	Palm Tree Theatres	Tampa
19	Cineplex Central	Orlando
20	Valley View Cinemas	Cleveland
21	Midtown Movies	St. Louis
22	Pacific Theatres	Kansas City
23	Sunrise Cinemas	Charlotte
24	Harborfront Movies	Indianapolis
25	Theatre Royale	Austin

4. Showtimes Table:

showtime_id (Primary Key), movie_id (Foreign Key referencing movies. movie_id), theater_id (Foreign Key referencing theaters. theater_id), start_time, end_time

Showtimes				
showtime_id	**movie_id**	**theater_id**	**start_time**	**end_time**
1	101	1	2/6/2024 15:00	2/6/2024 17:00
2	102	2	2/6/2024 18:30	2/6/2024 20:30
3	103	3	2/6/2024 14:00	2/6/2024 16:00
4	104	4	2/6/2024 17:45	2/6/2024 19:45
5	105	5	2/6/2024 20:00	2/6/2024 22:00
6	106	6	2/6/2024 15:15	2/6/2024 17:15
7	107	7	2/6/2024 18:00	2/6/2024 20:00
8	108	8	2/6/2024 16:30	2/6/2024 18:30
9	109	9	2/6/2024 19:00	2/6/2024 21:00
10	110	10	2/6/2024 14:45	2/6/2024 16:45
11	111	11	2/6/2024 17:30	2/6/2024 19:30
12	112	12	2/6/2024 20:15	2/6/2024 22:15
13	113	13	2/6/2024 15:45	2/6/2024 17:45
14	114	14	2/6/2024 18:45	2/6/2024 20:45
15	115	15	2/6/2024 16:00	2/6/2024 18:00
16	116	16	2/6/2024 19:15	2/6/2024 21:15
17	117	17	2/6/2024 14:30	2/6/2024 16:30
18	118	18	2/6/2024 17:00	2/6/2024 19:00
19	119	19	2/6/2024 19:30	2/6/2024 21:30
20	120	20	2/6/2024 15:30	2/6/2024 17:30
21	121	21	2/6/2024 18:15	2/6/2024 20:15
22	122	22	2/6/2024 16:45	2/6/2024 18:45

23	123	23	2/6/2024 19:45	2/6/2024 21:45
24	124	24	2/6/2024 14:15	2/6/2024 16:15
25	125	25	2/6/2024 17:45	2/6/2024 19:45

5. Bookings Table: -

booking_id (Primary Key), user_id (Foreign Key referencing users.user_id), showtime_id (Foreign Key referencing showtimes. showtime_id), num_tickets

Bookings			
booking_id	**user_id**	**showtime_id**	**num_tickets**
1	1	1	2
2	2	2	3
3	3	3	1
4	4	4	4
5	5	5	2
6	6	6	3
7	7	7	2
8	8	8	3
9	9	9	1
10	10	10	4
11	11	11	2
12	12	12	3
13	13	13	2
14	14	14	1
15	15	15	4
16	16	16	3
17	17	17	2
18	18	18	3
19	19	19	1
20	20	20	4
21	21	21	2
22	22	22	3
23	23	23	2
24	24	24	1
25	25	25	4

Assignment 2

Questions: -

Let's consider the `movies` table for generating 10 questions: -

Question 1: Retrieve the titles of all movies in the database.

Question 2: Find the genres of movies released on or after a specific date.

Question 3: Identify the number of movies in each genre.

Question 4: List the movies released before a certain year.

Question 5: Find the most recent movie in the database.

Question 6: Count the total number of movies in the database.

Question 7: Display the titles and release dates of movies.

Question 8: Retrieve the titles of movies that belong to a specific genre.

Question 9: Identify the oldest movie in the database.

Question 10: List the titles of movies along with their respective genres.

These questions cover various aspects of querying the `movies` table, such as selecting specific columns, filtering based on conditions, and aggregating data.

Let's consider the relationship between the `movies` and `showtimes` tables for generating 10 questions: -

Question 1: Retrieve the titles and genres of all movies that have showtimes.

Question 2: Find the theaters where a specific movie is currently being shown.

Question 3: Identify the number of showtimes for each movie.

Question 4: List the movies playing in a particular theater.

Question 5: Find the start times of all showtimes for a specific movie.

Question 6: What are the showtimes for the movie " The Matrix" at the theater named " Cinema Paradise"

Question 7: What is the username of the user who made the booking with ID is 2

Question 8: Identify the theaters with the most showtimes.

Question 9: List the movies along with the corresponding theater locations.

Question 10: Find the total number of tickets booked for each movie.

These questions involve queries that require joining the `movies` and `showtimes` tables to gather information from both tables.

Let's consider a relationship between the `movies`, `showtimes`, and `bookings` tables for generating 10 questions:

Question 1: Retrieve the titles and genres of all movies for which bookings have been made.

Question 2: Find the usernames and email addresses of users who booked tickets for a specific movie.

Question 3: Identify the total number of tickets booked for each movie.

Question 4: List the movies, showtimes, and corresponding theater names for a specific date.

Question 5: Find the total number of tickets booked by a specific user.

Question 6: Display the movie titles, showtimes, and booking details for a specific user.

Question 7: Retrieve the total revenue generated by each movie.

Question 8: Find the users who booked tickets for a movie in a specific genre.

Question 9: List the movies, showtimes, and booking details for a specific theater.

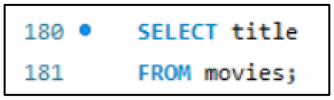
Question 10: Identify the most popular showtime (maximum bookings) for each movie.

These questions involve querying information from all three tables by performing joins and aggregations.

Assignment 3

Question 1: Retrieve the titles of all movies in the database.

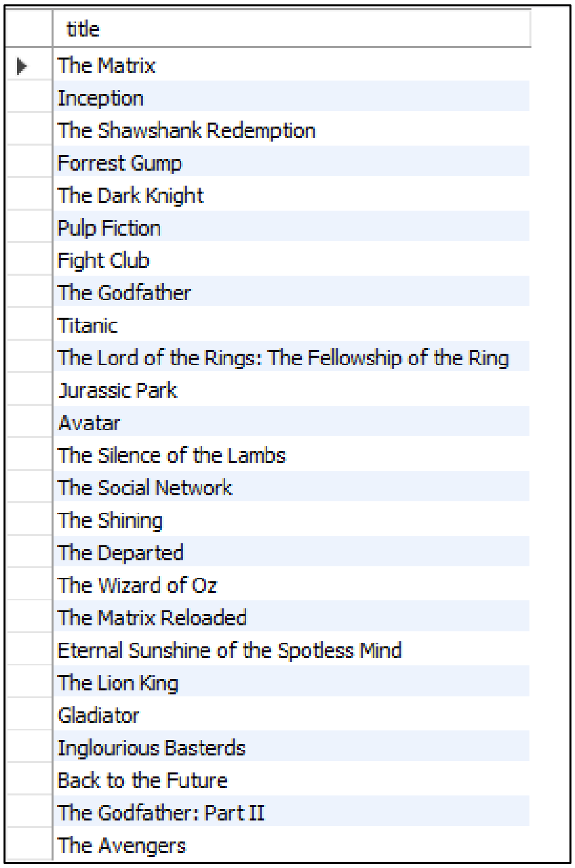
Ans: - To retrieve the titles of all movies in the database, you can use a simple SELECT query on the `movies` table.



```sql
SELECT title
FROM movies;
```

This query selects the `title` column from the `movies` table, returning a list of all movie titles in your database.

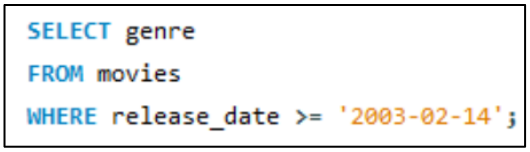
Output: -



title
The Matrix
Inception
The Shawshank Redemption
Forrest Gump
The Dark Knight
Pulp Fiction
Fight Club
The Godfather
Titanic
The Lord of the Rings: The Fellowship of the Ring
Jurassic Park
Avatar
The Silence of the Lambs
The Social Network
The Shining
The Departed
The Wizard of Oz
The Matrix Reloaded
Eternal Sunshine of the Spotless Mind
The Lion King
Gladiator
Inglourious Basterds
Back to the Future
The Godfather: Part II
The Avengers

Question 2: Find the genres of movies released on or after a specific date.

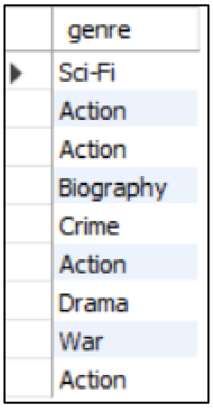
Ans:- To find the genres of movies released on or after a specific date, you can use a SELECT query with a WHERE clause to filter based on the release date.



```
SELECT genre
FROM movies
WHERE release_date >= '2003-02-14';
```

This query retrieves the genres of movies that were released on or after the specified date from the `movies` table.

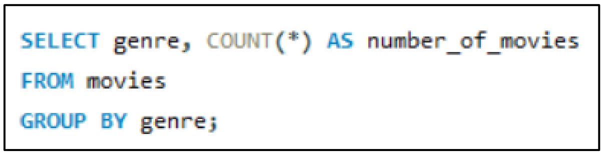
Output: -



genre
Sci-Fi
Action
Action
Biography
Crime
Action
Drama
War
Action

Question 3: Identify the number of movies in each genre.

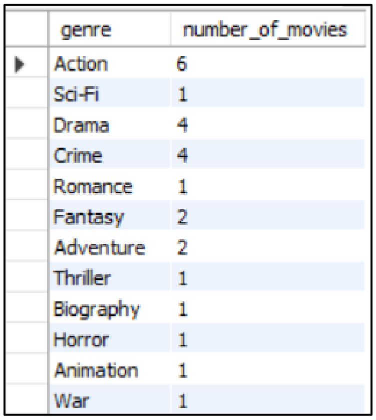
Ans: - To identify the number of movies in each genre, you can use the COUNT() function along with GROUP BY.



```
SELECT genre, COUNT(*) AS number_of_movies
FROM movies
GROUP BY genre;
```

This query counts the number of movies in each genre and presents the result with two columns: `genre` and `number_of_movies`.

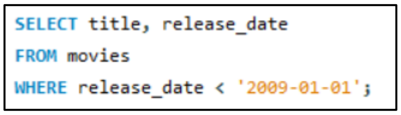
Output: -



genre	number_of_movies
Action	6
Sci-Fi	1
Drama	4
Crime	4
Romance	1
Fantasy	2
Adventure	2
Thriller	1
Biography	1
Horror	1
Animation	1
War	1

Question 4: List the movies released before a certain year.

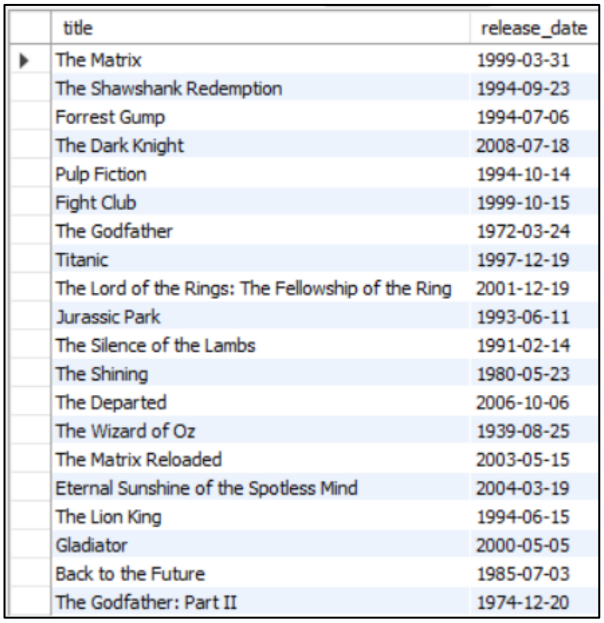
Ans:- To list the movies released before a certain year, you can use a simple SELECT query with a WHERE clause.



```
SELECT title, release_date
FROM movies
WHERE release_date < '2009-01-01';
```

This query retrieves the title and release date of movies that were released before the specified year from the `movies` table.

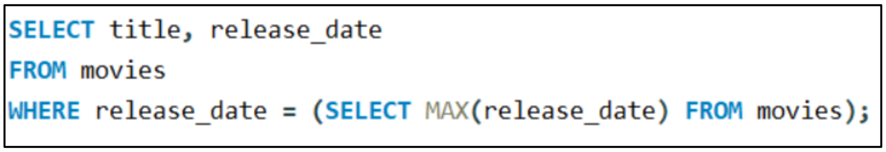
Output: -



title	release_date
The Matrix	1999-03-31
The Shawshank Redemption	1994-09-23
Forrest Gump	1994-07-06
The Dark Knight	2008-07-18
Pulp Fiction	1994-10-14
Fight Club	1999-10-15
The Godfather	1972-03-24
Titanic	1997-12-19
The Lord of the Rings: The Fellowship of the Ring	2001-12-19
Jurassic Park	1993-06-11
The Silence of the Lambs	1991-02-14
The Shining	1980-05-23
The Departed	2006-10-06
The Wizard of Oz	1939-08-25
The Matrix Reloaded	2003-05-15
Eternal Sunshine of the Spotless Mind	2004-03-19
The Lion King	1994-06-15
Gladiator	2000-05-05
Back to the Future	1985-07-03
The Godfather: Part II	1974-12-20

Question 5: Find the most recent movie in the database.

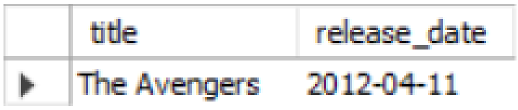
Ans:- To find the most recent movie in the database, you can use the MAX() function on the `release_date` column.



```
SELECT title, release_date
FROM movies
WHERE release_date = (SELECT MAX(release_date) FROM movies);
```

This query selects the title and release date of the movie(s) with the latest release date from the `movies` table.

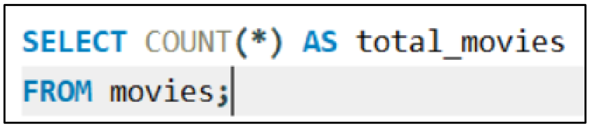
Output: -



title	release_date
The Avengers	2012-04-11

Question 6: Count the total number of movies in the database.

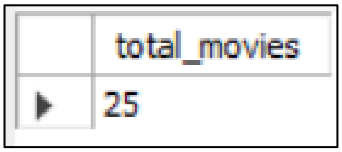
Ans:- To count the total number of movies in the database, you can use the COUNT() function.



```sql
SELECT COUNT(*) AS total_movies
FROM movies;
```

This query counts the total number of rows in the `movies` table, effectively giving you the total number of movies in the database. The result will be a single column named `total_movies`.

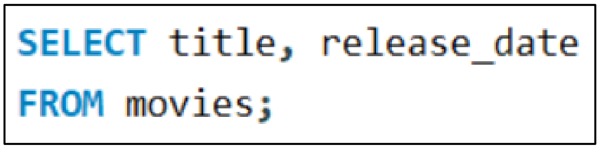
Output: -



	total_movies
▶	25

Question 7: Display the titles and release dates of movies.

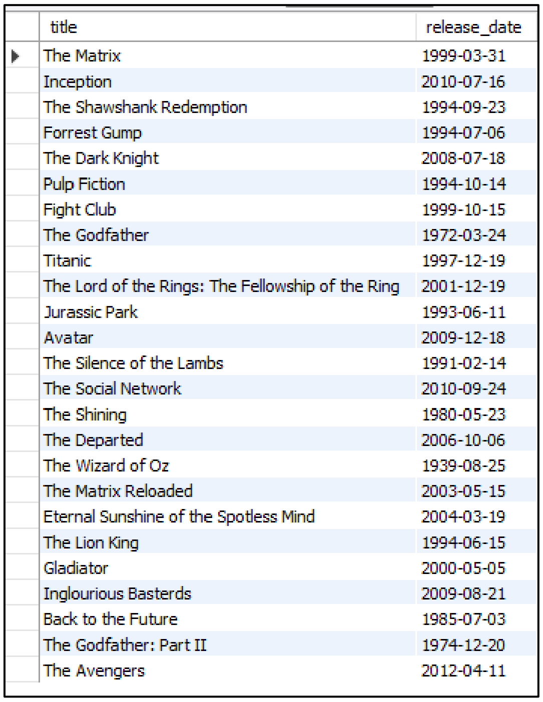
Ans:- To display the titles and release dates of movies.



```sql
SELECT title, release_date
FROM movies;
```

This query retrieves the `title` and `release_date` columns from the `movies` table.

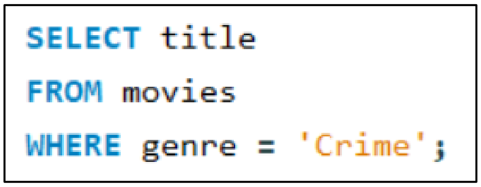
Output: -



	title	release_date
▶	The Matrix	1999-03-31
	Inception	2010-07-16
	The Shawshank Redemption	1994-09-23
	Forrest Gump	1994-07-06
	The Dark Knight	2008-07-18
	Pulp Fiction	1994-10-14
	Fight Club	1999-10-15
	The Godfather	1972-03-24
	Titanic	1997-12-19
	The Lord of the Rings: The Fellowship of the Ring	2001-12-19
	Jurassic Park	1993-06-11
	Avatar	2009-12-18
	The Silence of the Lambs	1991-02-14
	The Social Network	2010-09-24
	The Shining	1980-05-23
	The Departed	2006-10-06
	The Wizard of Oz	1939-08-25
	The Matrix Reloaded	2003-05-15
	Eternal Sunshine of the Spotless Mind	2004-03-19
	The Lion King	1994-06-15
	Gladiator	2000-05-05
	Inglourious Basterds	2009-08-21
	Back to the Future	1985-07-03
	The Godfather: Part II	1974-12-20
	The Avengers	2012-04-11

Question 8: Retrieve the titles of movies that belong to a specific genre.

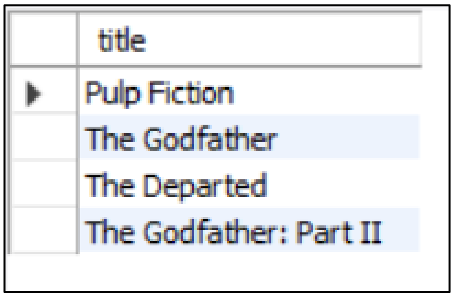
Ans:- To retrieve the titles of movies that belong to a specific genre, you can use a SELECT query with a WHERE clause to filter based on the genre.



```
SELECT title
FROM movies
WHERE genre = 'Crime';
```

This query retrieves the titles of movies from the `movies` table that belong to the specified genre.

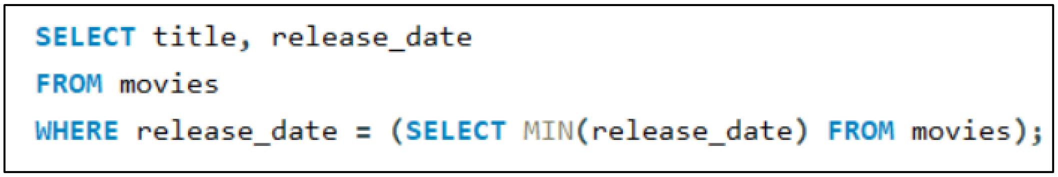
Output: -



	title
▶	Pulp Fiction
	The Godfather
	The Departed
	The Godfather: Part II

Question 9: Identify the oldest movie in the database.

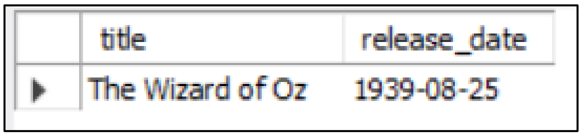
Ans:- To identify the oldest movie in the database, you can use the MIN() function on the `release_date` column.



```
SELECT title, release_date
FROM movies
WHERE release_date = (SELECT MIN(release_date) FROM movies);
```

This query selects the title and release date of the movie(s) with the earliest release date from the `movies` table.

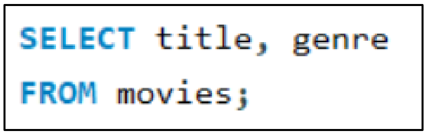
Output: -



	title	release_date
▶	The Wizard of Oz	1939-08-25

Question 10: List the titles of movies along with their respective genres.

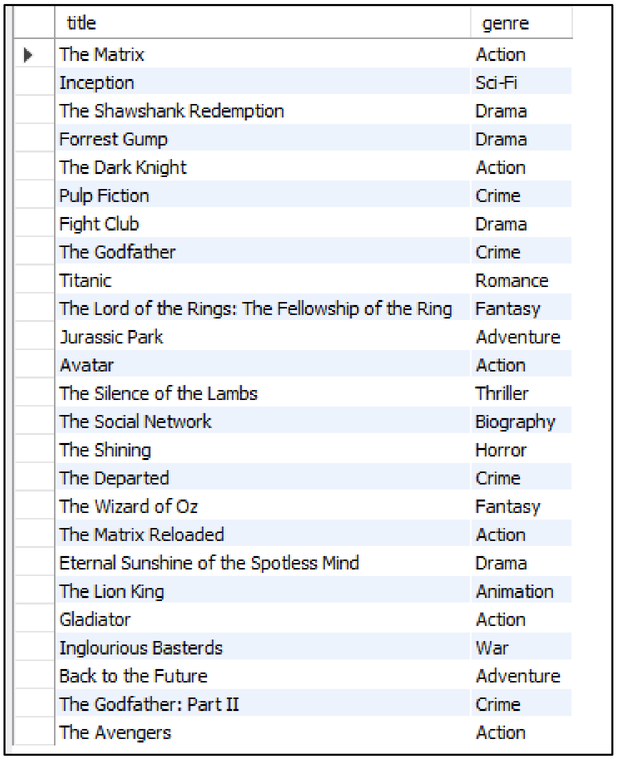
Ans:- To list the titles of movies along with their respective genres.



```
SELECT title, genre
FROM movies;
```

This query retrieves the `title` and `genre` columns from the `movies` table.

Output: -

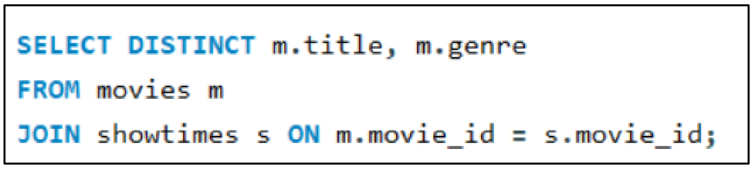


	title	genre
▶	The Matrix	Action
	Inception	Sci-Fi
	The Shawshank Redemption	Drama
	Forrest Gump	Drama
	The Dark Knight	Action
	Pulp Fiction	Crime
	Fight Club	Drama
	The Godfather	Crime
	Titanic	Romance
	The Lord of the Rings: The Fellowship of the Ring	Fantasy
	Jurassic Park	Adventure
	Avatar	Action
	The Silence of the Lambs	Thriller
	The Social Network	Biography
	The Shining	Horror
	The Departed	Crime
	The Wizard of Oz	Fantasy
	The Matrix Reloaded	Action
	Eternal Sunshine of the Spotless Mind	Drama
	The Lion King	Animation
	Gladiator	Action
	Inglourious Basterds	War
	Back to the Future	Adventure
	The Godfather: Part II	Crime
	The Avengers	Action

Assignment 4

Question 1: Retrieve the titles and genres of all movies that have showtimes.

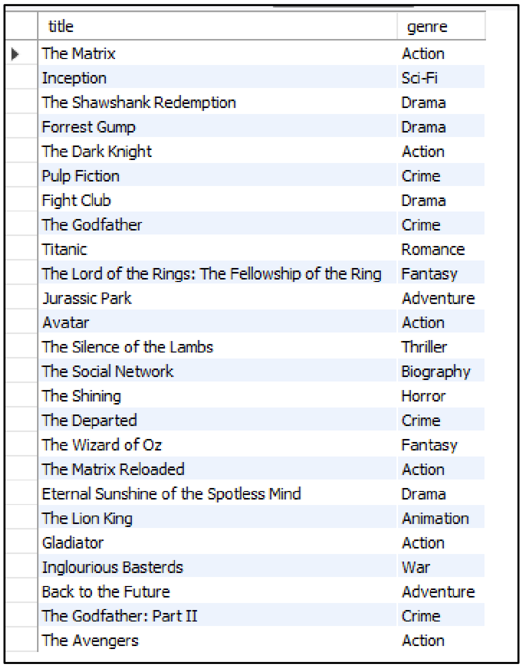
Ans:- To retrieve the titles and genres of all movies that have showtimes, you can use a SELECT query with a JOIN clause to combine information from the `movies` and `showtimes` tables.



```
SELECT DISTINCT m.title, m.genre
FROM movies m
JOIN showtimes s ON m.movie_id = s.movie_id;
```

This query selects distinct movie titles and genres from the `movies` table for movies that have associated showtimes in the `showtimes` table.

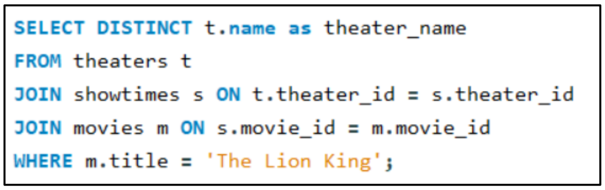
Output: -



title	genre
The Matrix	Action
Inception	Sci-Fi
The Shawshank Redemption	Drama
Forrest Gump	Drama
The Dark Knight	Action
Pulp Fiction	Crime
Fight Club	Drama
The Godfather	Crime
Titanic	Romance
The Lord of the Rings: The Fellowship of the Ring	Fantasy
Jurassic Park	Adventure
Avatar	Action
The Silence of the Lambs	Thriller
The Social Network	Biography
The Shining	Horror
The Departed	Crime
The Wizard of Oz	Fantasy
The Matrix Reloaded	Action
Eternal Sunshine of the Spotless Mind	Drama
The Lion King	Animation
Gladiator	Action
Inglourious Basterds	War
Back to the Future	Adventure
The Godfather: Part II	Crime
The Avengers	Action

Question 2: Find the theaters where a specific movie is currently being shown.

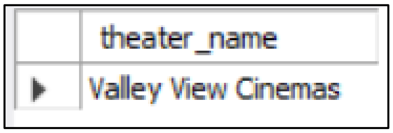
Ans:- To find the theaters where a specific movie is currently being shown, you can use a SELECT query with a JOIN clause involving the `movies`, `showtimes`, and `theaters` tables.



```
SELECT DISTINCT t.name as theater_name
FROM theaters t
JOIN showtimes s ON t.theater_id = s.theater_id
JOIN movies m ON s.movie_id = m.movie_id
WHERE m.title = 'The Lion King';
```

Replace `'specific_movie_title'` with the actual title of the movie you're interested in. This query retrieves the distinct names of theaters where the specified movie is currently being shown.

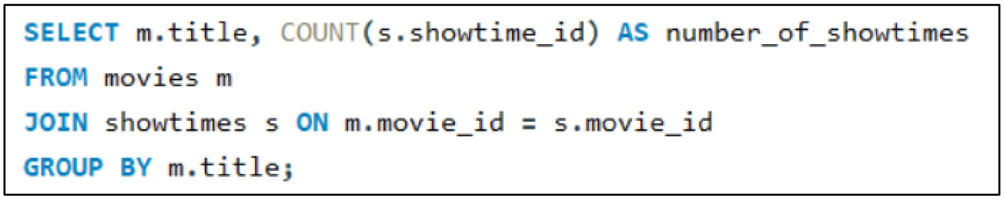
Output: -



theater_name
Valley View Cinemas

Question 3: Identify the number of showtimes for each movie.

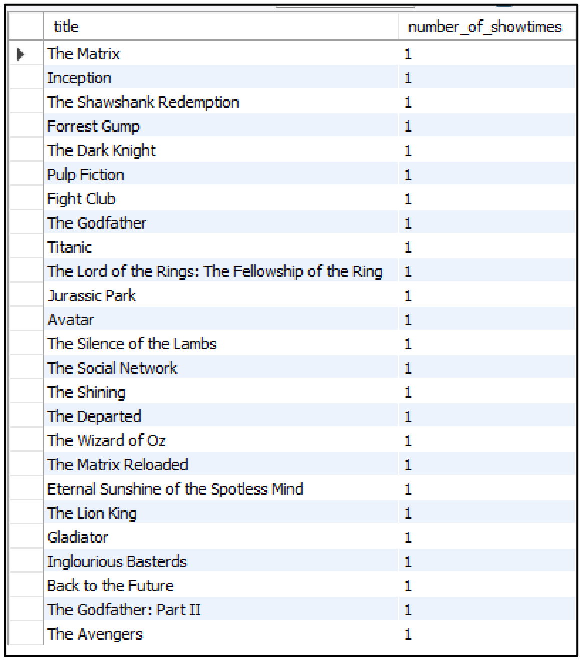
Ans:- To identify the number of showtimes for each movie, you can use a SELECT query with a JOIN and GROUP BY clause.



```
SELECT m.title, COUNT(s.showtime_id) AS number_of_showtimes
FROM movies m
JOIN showtimes s ON m.movie_id = s.movie_id
GROUP BY m.title;
```

This query retrieves the movie title and counts the number of showtimes for each movie using the `COUNT` function. The result will show the title of each movie along with the corresponding number of showtimes. Adjust the column and table names based on your actual database schema if needed.

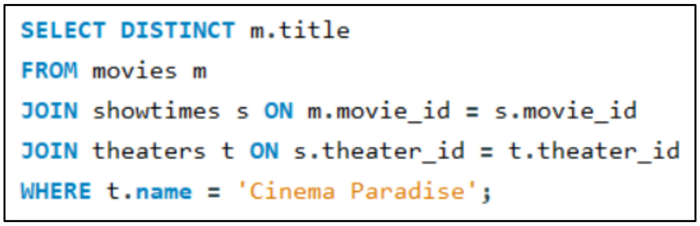
Output: -



title	number_of_showtimes
The Matrix	1
Inception	1
The Shawshank Redemption	1
Forrest Gump	1
The Dark Knight	1
Pulp Fiction	1
Fight Club	1
The Godfather	1
Titanic	1
The Lord of the Rings: The Fellowship of the Ring	1
Jurassic Park	1
Avatar	1
The Silence of the Lambs	1
The Social Network	1
The Shining	1
The Departed	1
The Wizard of Oz	1
The Matrix Reloaded	1
Eternal Sunshine of the Spotless Mind	1
The Lion King	1
Gladiator	1
Inglourious Basterds	1
Back to the Future	1
The Godfather: Part II	1
The Avengers	1

Question 4: List the movies playing in a particular theater.

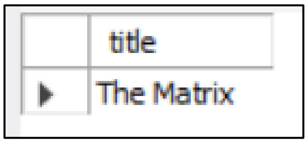
Ans:- To list the movies playing in a particular theater, you can use a SELECT query with a JOIN clause involving the `movies`, `showtimes`, and `theaters` tables.



```sql
SELECT DISTINCT m.title
FROM movies m
JOIN showtimes s ON m.movie_id = s.movie_id
JOIN theaters t ON s.theater_id = t.theater_id
WHERE t.name = 'Cinema Paradise';
```

Replace `'specific_theater_name'` with the actual name of the theater you're interested in. This query retrieves the distinct titles of movies playing in the specified theater.

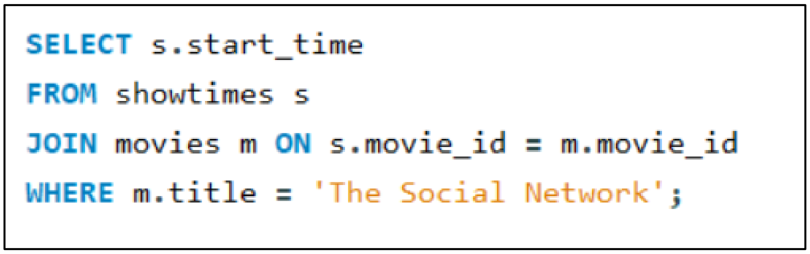
Output: -



	title
▶	The Matrix

Question 5: Find the start times of all showtimes for a specific movie.

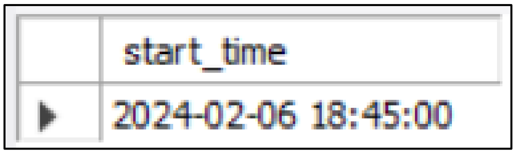
Ans:- To find the start times of all showtimes for a specific movie, you can use a SELECT query with a JOIN clause involving the `movies` and `showtimes` tables.



```sql
SELECT s.start_time
FROM showtimes s
JOIN movies m ON s.movie_id = m.movie_id
WHERE m.title = 'The Social Network';
```

Replace `'specific_movie_title'` with the actual title of the movie you're interested in. This query retrieves the start times of all showtimes for the specified movie. Adjust the column and table names based on your actual database schema if needed.

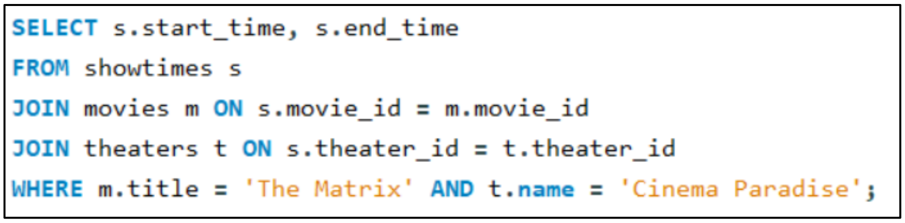
Output: -



	start_time
▶	2024-02-06 18:45:00

Question 6: What are the showtimes for the movie " The Matrix" at the theater named " Cinema Paradise"

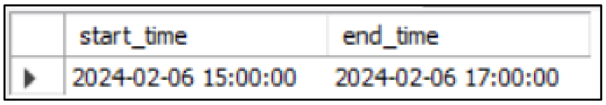
Ans: - To retrieve the showtimes for the movie " The Matrix " at the theater named " Cinema Paradise," you would need to perform a JOIN operation between the Movies table and the Showtimes table, and another JOIN operation between the Theaters table and the Showtimes table, filtering the results based on the movie title and theater name.



```
SELECT s.start_time, s.end_time
FROM showtimes s
JOIN movies m ON s.movie_id = m.movie_id
JOIN theaters t ON s.theater_id = t.theater_id
WHERE m.title = 'The Matrix' AND t.name = 'Cinema Paradise';
```

This query retrieves the start time and end time from the Showtimes table, where the movie title is " The Matrix " and the theater name is " Cinema Paradise ".

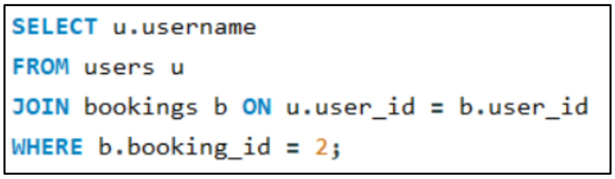
Output: -



	start_time	end_time
▶	2024-02-06 15:00:00	2024-02-06 17:00:00

Question 7: What is the username of the user who made the booking with ID is 2

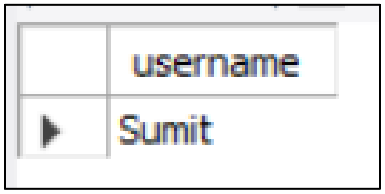
Ans: - To retrieve the username of the user who made the booking with ID 2, you can perform a JOIN operation between the Users table and the Bookings table, filtering the results based on the booking ID.



```
SELECT u.username
FROM users u
JOIN bookings b ON u.user_id = b.user_id
WHERE b.booking_id = 2;
```

This query selects the username from the Users table where the user ID in the Users table matches the user ID associated with the booking ID 2 in the Bookings table.

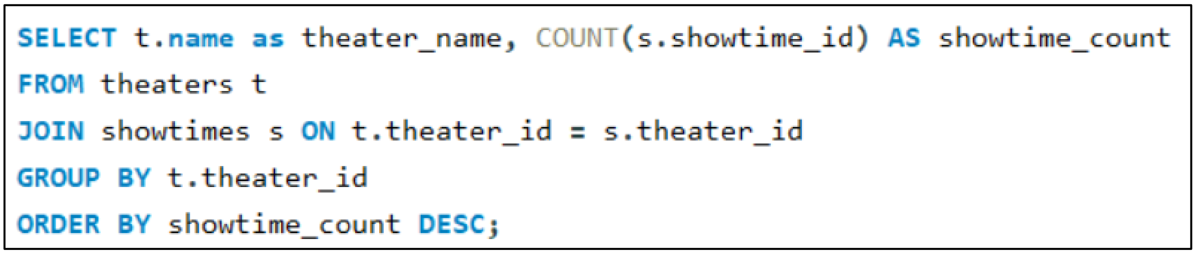
Output: -



	username
▶	Sumit

Question 8: Identify the theaters with the most showtimes.

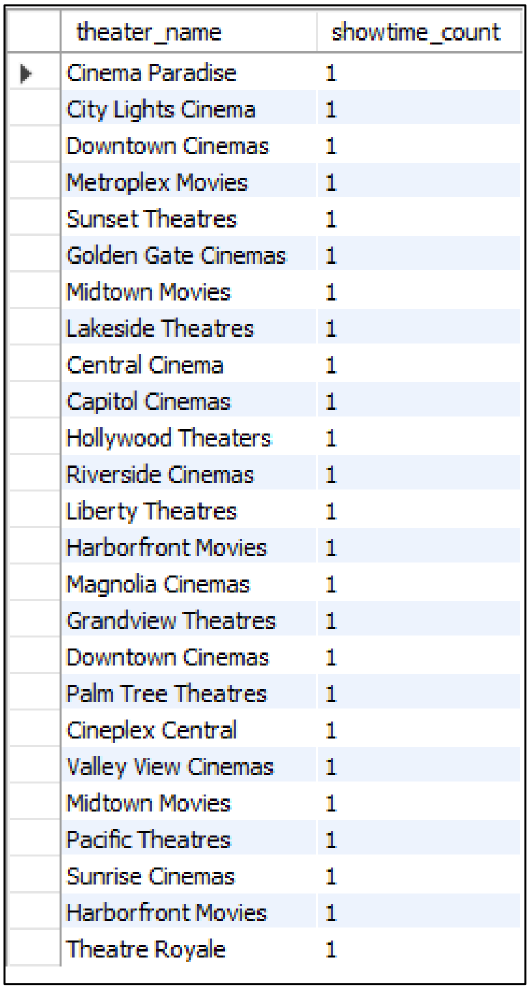
Ans:- To identify the theaters with the most showtimes, you can use a SELECT query with a JOIN, GROUP BY, and ORDER BY clauses.



```
SELECT t.name as theater_name, COUNT(s.showtime_id) AS showtime_count
FROM theaters t
JOIN showtimes s ON t.theater_id = s.theater_id
GROUP BY t.theater_id
ORDER BY showtime_count DESC;
```

This query counts the number of showtimes for each theater, groups the results by theater, and then orders the result in descending order based on the showtime count. The theater with the most showtimes will appear at the top of the result set.

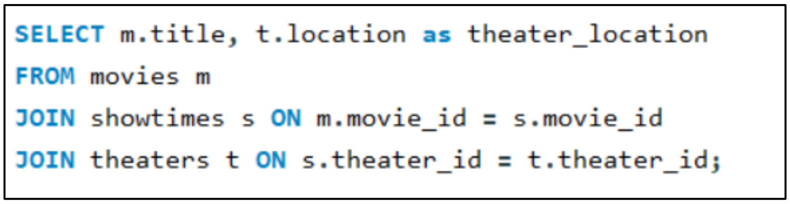
Output: -



theater_name	showtime_count
Cinema Paradise	1
City Lights Cinema	1
Downtown Cinemas	1
Metroplex Movies	1
Sunset Theatres	1
Golden Gate Cinemas	1
Midtown Movies	1
Lakeside Theatres	1
Central Cinema	1
Capitol Cinemas	1
Hollywood Theaters	1
Riverside Cinemas	1
Liberty Theatres	1
Harborfront Movies	1
Magnolia Cinemas	1
Grandview Theatres	1
Downtown Cinemas	1
Palm Tree Theatres	1
Cineplex Central	1
Valley View Cinemas	1
Midtown Movies	1
Pacific Theatres	1
Sunrise Cinemas	1
Harborfront Movies	1
Theatre Royale	1

Question 9: List the movies along with the corresponding theater locations.

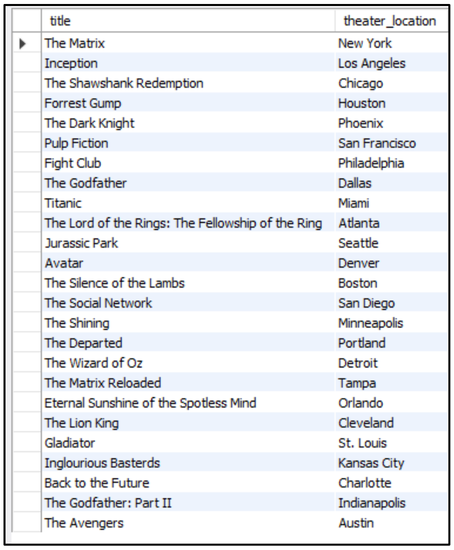
Ans:- To list the movies along with the corresponding theater locations, you can use a SELECT query with a JOIN clause involving the `movies`, `showtimes`, and `theaters` tables.



```
SELECT m.title, t.location as theater_location
FROM movies m
JOIN showtimes s ON m.movie_id = s.movie_id
JOIN theaters t ON s.theater_id = t.theater_id;
```

This query retrieves the movie titles and theater locations by joining the `movies`, `showtimes`, and `theaters` tables.

Output: -



title	theater_location
The Matrix	New York
Inception	Los Angeles
The Shawshank Redemption	Chicago
Forrest Gump	Houston
The Dark Knight	Phoenix
Pulp Fiction	San Francisco
Fight Club	Philadelphia
The Godfather	Dallas
Titanic	Miami
The Lord of the Rings: The Fellowship of the Ring	Atlanta
Jurassic Park	Seattle
Avatar	Denver
The Silence of the Lambs	Boston
The Social Network	San Diego
The Shining	Minneapolis
The Departed	Portland
The Wizard of Oz	Detroit
The Matrix Reloaded	Tampa
Eternal Sunshine of the Spotless Mind	Orlando
The Lion King	Cleveland
Gladiator	St. Louis
Inglourious Basterds	Kansas City
Back to the Future	Charlotte
The Godfather: Part II	Indianapolis
The Avengers	Austin

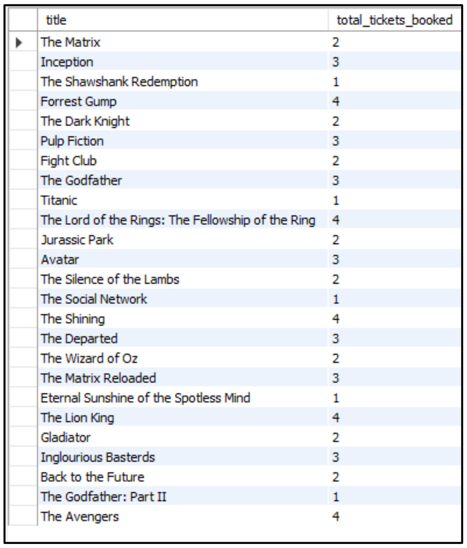
Question 10: Find the total number of tickets booked for each movie.

Ans:- To find the total number of tickets booked for each movie, you can use a SELECT query with a JOIN, GROUP BY, and SUM clause. Assuming there's a `bookings` table that stores information about each booking.

```
SELECT m.title, SUM(b.num_tickets) AS total_tickets_booked
FROM movies m
JOIN showtimes s ON m.movie_id = s.movie_id
JOIN bookings b ON s.showtime_id = b.showtime_id
GROUP BY m.title;
```

This query joins the `movies`, `showtimes`, and `bookings` tables, groups the result by movie title, and calculates the total number of tickets booked for each movie.

Output: -

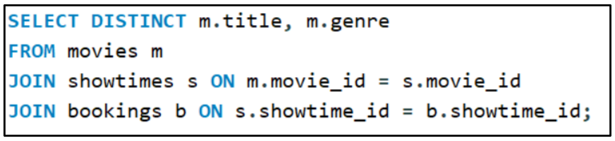


title	total_tickets_booked
The Matrix	2
Inception	3
The Shawshank Redemption	1
Forrest Gump	4
The Dark Knight	2
Pulp Fiction	3
Fight Club	2
The Godfather	3
Titanic	1
The Lord of the Rings: The Fellowship of the Ring	4
Jurassic Park	2
Avatar	3
The Silence of the Lambs	2
The Social Network	1
The Shining	4
The Departed	3
The Wizard of Oz	2
The Matrix Reloaded	3
Eternal Sunshine of the Spotless Mind	1
The Lion King	4
Gladiator	2
Inglourious Basterds	3
Back to the Future	2
The Godfather: Part II	1
The Avengers	4

Assignment 5

Question 1: Retrieve the titles and genres of all movies for which bookings have been made.

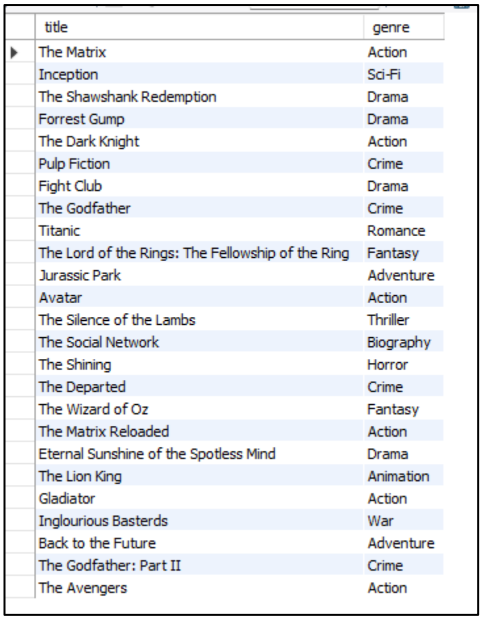
Ans: - To retrieve the titles and genres of all movies for which bookings have been made, you can use a SELECT query with a JOIN clause involving the `movies`, `showtimes`, and `bookings` tables.



```sql
SELECT DISTINCT m.title, m.genre
FROM movies m
JOIN showtimes s ON m.movie_id = s.movie_id
JOIN bookings b ON s.showtime_id = b.showtime_id;
```

This query retrieves the distinct titles and genres of movies for which bookings have been made by joining the `movies`, `showtimes`, and `bookings` tables.

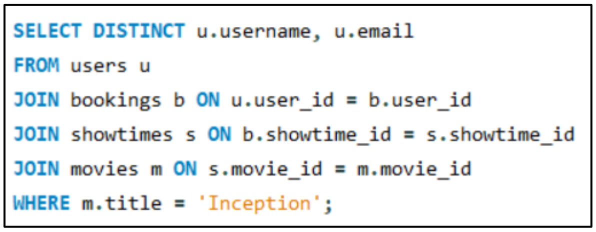
Output: -



title	genre
The Matrix	Action
Inception	Sci-Fi
The Shawshank Redemption	Drama
Forrest Gump	Drama
The Dark Knight	Action
Pulp Fiction	Crime
Fight Club	Drama
The Godfather	Crime
Titanic	Romance
The Lord of the Rings: The Fellowship of the Ring	Fantasy
Jurassic Park	Adventure
Avatar	Action
The Silence of the Lambs	Thriller
The Social Network	Biography
The Shining	Horror
The Departed	Crime
The Wizard of Oz	Fantasy
The Matrix Reloaded	Action
Eternal Sunshine of the Spotless Mind	Drama
The Lion King	Animation
Gladiator	Action
Inglourious Basterds	War
Back to the Future	Adventure
The Godfather: Part II	Crime
The Avengers	Action

Question 2: Find the usernames and email addresses of users who booked tickets for a specific movie.

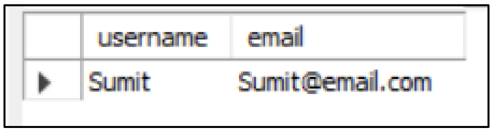
Ans:- To find the usernames and email addresses of users who booked tickets for a specific movie, you can use a SELECT query with a JOIN clause involving the `users`, `bookings`, `showtimes`, and `movies` tables.



```sql
SELECT DISTINCT u.username, u.email
FROM users u
JOIN bookings b ON u.user_id = b.user_id
JOIN showtimes s ON b.showtime_id = s.showtime_id
JOIN movies m ON s.movie_id = m.movie_id
WHERE m.title = 'Inception';
```

Replace `'specific_movie_title'` with the actual title of the movie you're interested in. This query retrieves the distinct usernames and email addresses of users who booked tickets for the specified movie.

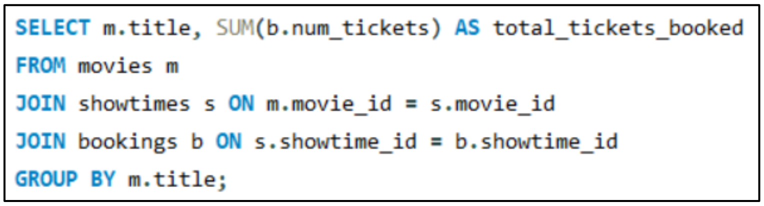
Output: -



username	email
Sumit	Sumit@email.com

Question 3: Identify the total number of tickets booked for each movie.

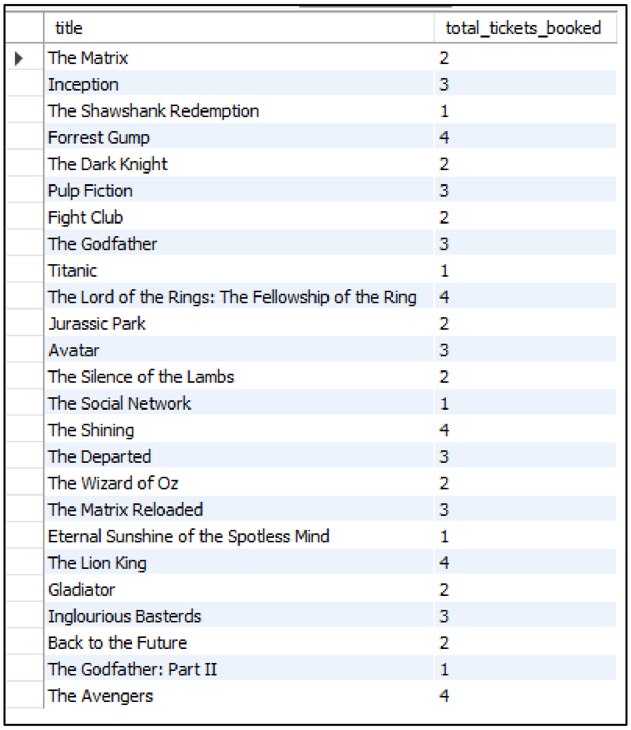
Ans:- To identify the total number of tickets booked for each movie, you can use a SELECT query with a JOIN, GROUP BY, and SUM clause. Assuming there's a `bookings` table that stores information about each booking.



```
SELECT m.title, SUM(b.num_tickets) AS total_tickets_booked
FROM movies m
JOIN showtimes s ON m.movie_id = s.movie_id
JOIN bookings b ON s.showtime_id = b.showtime_id
GROUP BY m.title;
```

This query joins the `movies`, `showtimes`, and `bookings` tables, groups the result by movie title, and calculates the total number of tickets booked for each movie. Adjust the column and table names based on your actual database schema if needed.

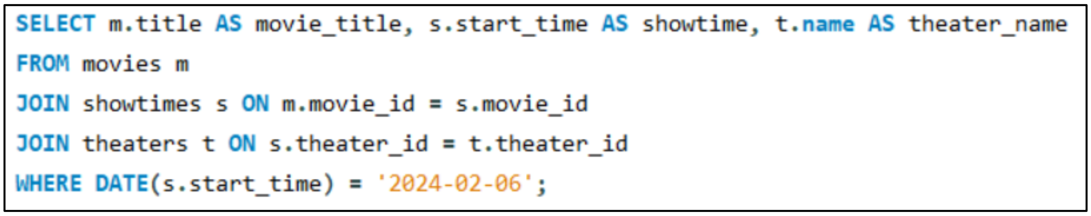
Output: -



title	total_tickets_booked
The Matrix	2
Inception	3
The Shawshank Redemption	1
Forrest Gump	4
The Dark Knight	2
Pulp Fiction	3
Fight Club	2
The Godfather	3
Titanic	1
The Lord of the Rings: The Fellowship of the Ring	4
Jurassic Park	2
Avatar	3
The Silence of the Lambs	2
The Social Network	1
The Shining	4
The Departed	3
The Wizard of Oz	2
The Matrix Reloaded	3
Eternal Sunshine of the Spotless Mind	1
The Lion King	4
Gladiator	2
Inglourious Basterds	3
Back to the Future	2
The Godfather: Part II	1
The Avengers	4

Question 4: List the movies, showtimes, and corresponding theater names for a specific date.

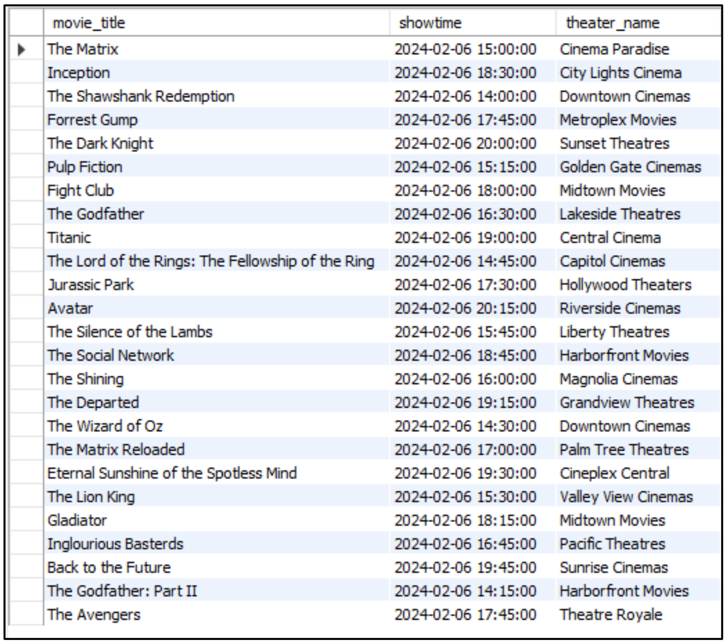
Ans:- To list the movies, showtimes, and corresponding theater names for a specific date, you can use a SELECT query with a JOIN clause involving the `movies`, `showtimes`, and `theaters` tables.



```
SELECT m.title AS movie_title, s.start_time AS showtime, t.name AS theater_name
FROM movies m
JOIN showtimes s ON m.movie_id = s.movie_id
JOIN theaters t ON s.theater_id = t.theater_id
WHERE DATE(s.start_time) = '2024-02-06';
```

Replace `'specific_date'` with the actual date you're interested in. This query retrieves the movie titles, showtimes, and corresponding theater names for the specified date.

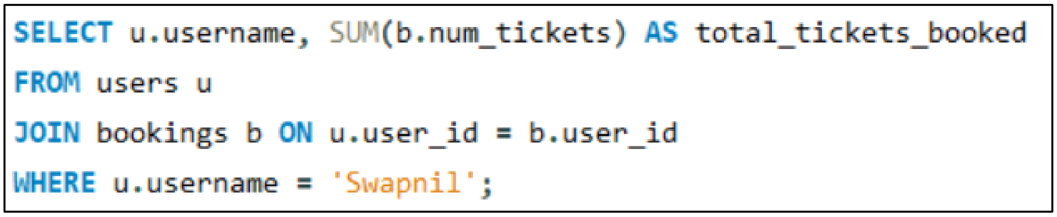
Output: -



movie_title	showtime	theater_name
The Matrix	2024-02-06 15:00:00	Cinema Paradise
Inception	2024-02-06 18:30:00	City Lights Cinema
The Shawshank Redemption	2024-02-06 14:00:00	Downtown Cinemas
Forrest Gump	2024-02-06 17:45:00	Metroplex Movies
The Dark Knight	2024-02-06 20:00:00	Sunset Theatres
Pulp Fiction	2024-02-06 15:15:00	Golden Gate Cinemas
Fight Club	2024-02-06 18:00:00	Midtown Movies
The Godfather	2024-02-06 16:30:00	Lakeside Theatres
Titanic	2024-02-06 19:00:00	Central Cinema
The Lord of the Rings: The Fellowship of the Ring	2024-02-06 14:45:00	Capitol Cinemas
Jurassic Park	2024-02-06 17:30:00	Hollywood Theaters
Avatar	2024-02-06 20:15:00	Riverside Cinemas
The Silence of the Lambs	2024-02-06 15:45:00	Liberty Theatres
The Social Network	2024-02-06 18:45:00	Harborfront Movies
The Shining	2024-02-06 16:00:00	Magnolia Cinemas
The Departed	2024-02-06 19:15:00	Grandview Theatres
The Wizard of Oz	2024-02-06 14:30:00	Downtown Cinemas
The Matrix Reloaded	2024-02-06 17:00:00	Palm Tree Theatres
Eternal Sunshine of the Spotless Mind	2024-02-06 19:30:00	Cineplex Central
The Lion King	2024-02-06 15:30:00	Valley View Cinemas
Gladiator	2024-02-06 18:15:00	Midtown Movies
Inglourious Basterds	2024-02-06 16:45:00	Pacific Theatres
Back to the Future	2024-02-06 19:45:00	Sunrise Cinemas
The Godfather: Part II	2024-02-06 14:15:00	Harborfront Movies
The Avengers	2024-02-06 17:45:00	Theatre Royale

Question 5: Find the total number of tickets booked by a specific user.

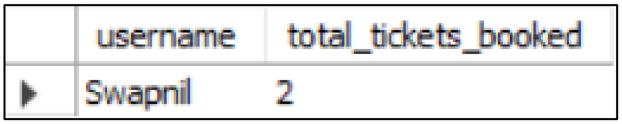
Ans:- To find the total number of tickets booked by a specific user, you can use a SELECT query with a WHERE clause to filter based on the user's information and then use the SUM function to calculate the total number of tickets. Assuming there's a `users` table storing user information and a `bookings` table for booking details.



```
SELECT u.username, SUM(b.num_tickets) AS total_tickets_booked
FROM users u
JOIN bookings b ON u.user_id = b.user_id
WHERE u.username = 'Swapnil';
```

Replace `'specific_username'` with the actual username of the user you're interested in. This query retrieves the total number of tickets booked by the specified user.

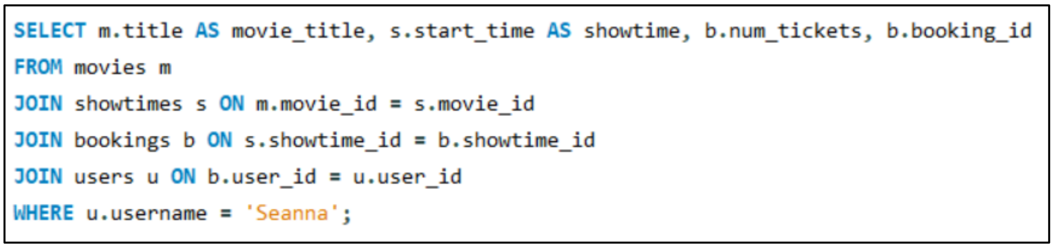
Output: -



username	total_tickets_booked
Swapnil	2

Question 6: Display the movie titles, showtimes, and booking details for a specific user.

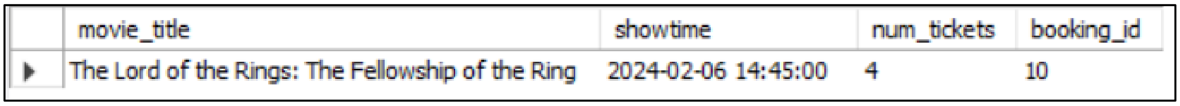
Ans:- To display the movie titles, showtimes, and booking details for a specific user, you can use a SELECT query with a JOIN clause involving the `movies`, `showtimes`, `bookings`, and `users` tables.



```
SELECT m.title AS movie_title, s.start_time AS showtime, b.num_tickets, b.booking_id
FROM movies m
JOIN showtimes s ON m.movie_id = s.movie_id
JOIN bookings b ON s.showtime_id = b.showtime_id
JOIN users u ON b.user_id = u.user_id
WHERE u.username = 'Seanna';
```

Replace `'specific_username'` with the actual username of the user you're interested in. This query retrieves the movie titles, showtimes, and booking details for the specified user.

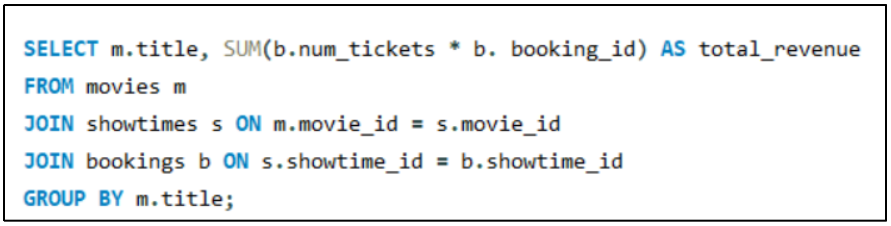
Output: -



movie_title	showtime	num_tickets	booking_id
The Lord of the Rings: The Fellowship of the Ring	2024-02-06 14:45:00	4	10

Question 7: Retrieve the total revenue generated by each movie.

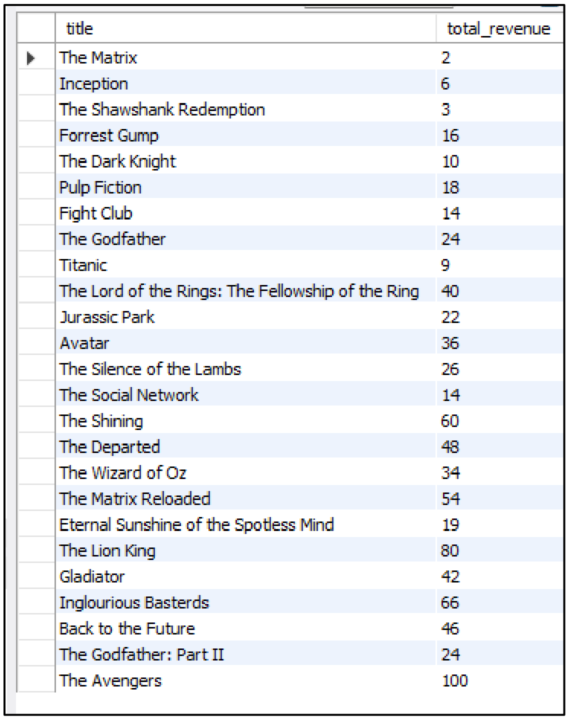
Ans:- To retrieve the total revenue generated by each movie, you can use a SELECT query with a JOIN, GROUP BY, and SUM clause. Assuming there's a `bookings` table that stores information about each booking and includes the price per ticket.



```
SELECT m.title, SUM(b.num_tickets * b. booking_id) AS total_revenue
FROM movies m
JOIN showtimes s ON m.movie_id = s.movie_id
JOIN bookings b ON s.showtime_id = b.showtime_id
GROUP BY m.title;
```

This query joins the `movies`, `showtimes`, and `bookings` tables, groups the result by movie title, and calculates the total revenue generated by each movie by multiplying the number of tickets with the ticket price and then summing them up.

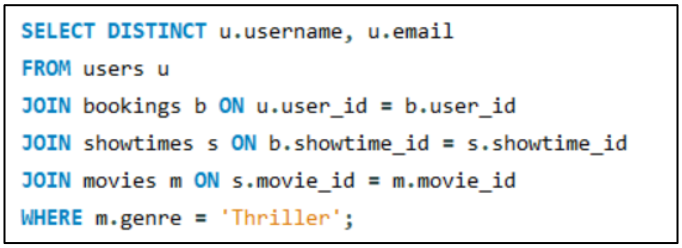
Output: -



title	total_revenue
The Matrix	2
Inception	6
The Shawshank Redemption	3
Forrest Gump	16
The Dark Knight	10
Pulp Fiction	18
Fight Club	14
The Godfather	24
Titanic	9
The Lord of the Rings: The Fellowship of the Ring	40
Jurassic Park	22
Avatar	36
The Silence of the Lambs	26
The Social Network	14
The Shining	60
The Departed	48
The Wizard of Oz	34
The Matrix Reloaded	54
Eternal Sunshine of the Spotless Mind	19
The Lion King	80
Gladiator	42
Inglourious Basterds	66
Back to the Future	46
The Godfather: Part II	24
The Avengers	100

Question 8: Find the users who booked tickets for a movie in a specific genre.

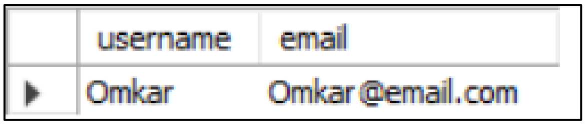
Ans:- To find the users who booked tickets for a movie in a specific genre, you can use a SELECT query with a JOIN clause involving the `users`, `bookings`, `showtimes`, and `movies` tables.



```
SELECT DISTINCT u.username, u.email
FROM users u
JOIN bookings b ON u.user_id = b.user_id
JOIN showtimes s ON b.showtime_id = s.showtime_id
JOIN movies m ON s.movie_id = m.movie_id
WHERE m.genre = 'Thriller';
```

Replace `'specific_genre'` with the actual genre you're interested in. This query retrieves the distinct usernames and email addresses of users who booked tickets for movies in the specified genre.

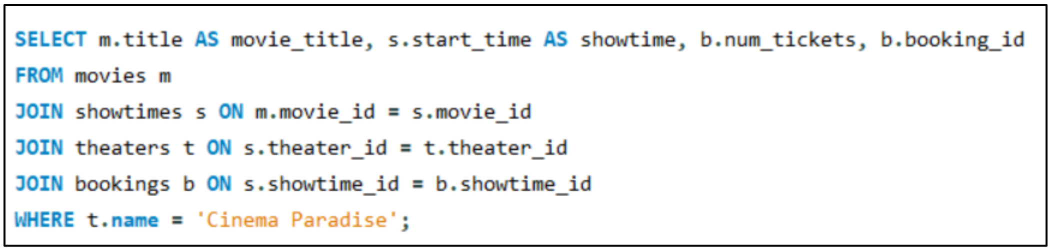
Output: -



username	email
Omkar	Omkar@email.com

Question 9: List the movies, showtimes, and booking details for a specific theater.

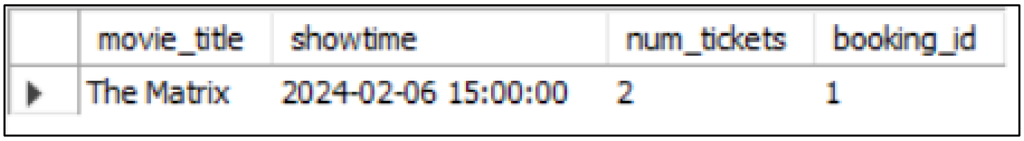
Ans:- To list the movies, showtimes, and booking details for a specific theater, you can use a SELECT query with a JOIN clause involving the `movies`, `showtimes`, `bookings`, and `theaters` tables.



```
SELECT m.title AS movie_title, s.start_time AS showtime, b.num_tickets, b.booking_id
FROM movies m
JOIN showtimes s ON m.movie_id = s.movie_id
JOIN theaters t ON s.theater_id = t.theater_id
JOIN bookings b ON s.showtime_id = b.showtime_id
WHERE t.name = 'Cinema Paradise';
```

Replace `'specific_theater_name'` with the actual name of the theater you're interested in. This query retrieves the movie titles, showtimes, and booking details for the specified theater.

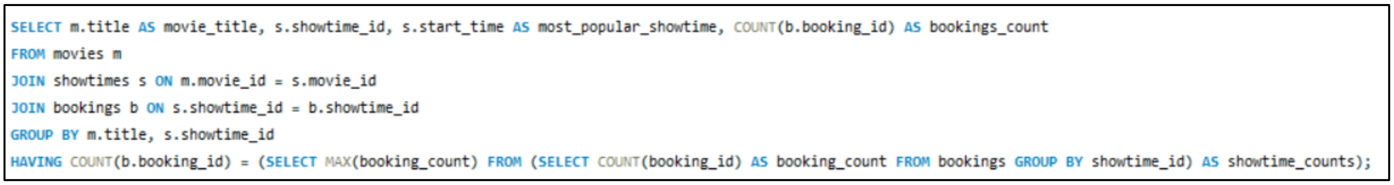
Output: -



movie_title	showtime	num_tickets	booking_id
The Matrix	2024-02-06 15:00:00	2	1

Question 10: Identify the most popular showtime (maximum bookings) for each movie.

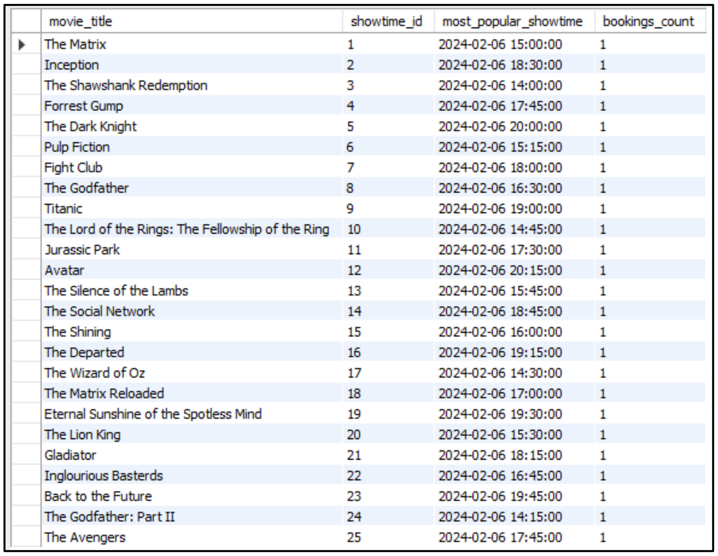
Ans:- To identify the most popular showtime (maximum bookings) for each movie, you can use a SELECT query with a JOIN, GROUP BY, and MAX clause.



```
SELECT m.title AS movie_title, s.showtime_id, s.start_time AS most_popular_showtime, COUNT(b.booking_id) AS bookings_count
FROM movies m
JOIN showtimes s ON m.movie_id = s.movie_id
JOIN bookings b ON s.showtime_id = b.showtime_id
GROUP BY m.title, s.showtime_id
HAVING COUNT(b.booking_id) = (SELECT MAX(booking_count) FROM (SELECT COUNT(booking_id) AS booking_count FROM bookings GROUP BY showtime_id) AS showtime_counts);
```

This query joins the `movies`, `showtimes`, and `bookings` tables, groups the result by movie title and showtime_id, calculates the number of bookings for each showtime, and selects only those showtimes with the maximum bookings count for each movie.

Output: -



movie_title	showtime_id	most_popular_showtime	bookings_count
The Matrix	1	2024-02-06 15:00:00	1
Inception	2	2024-02-06 18:30:00	1
The Shawshank Redemption	3	2024-02-06 14:00:00	1
Forrest Gump	4	2024-02-06 17:45:00	1
The Dark Knight	5	2024-02-06 20:00:00	1
Pulp Fiction	6	2024-02-06 15:15:00	1
Fight Club	7	2024-02-06 18:00:00	1
The Godfather	8	2024-02-06 16:30:00	1
Titanic	9	2024-02-06 19:00:00	1
The Lord of the Rings: The Fellowship of the Ring	10	2024-02-06 14:45:00	1
Jurassic Park	11	2024-02-06 17:30:00	1
Avatar	12	2024-02-06 20:15:00	1
The Silence of the Lambs	13	2024-02-06 15:45:00	1
The Social Network	14	2024-02-06 18:45:00	1
The Shining	15	2024-02-06 16:00:00	1
The Departed	16	2024-02-06 19:15:00	1
The Wizard of Oz	17	2024-02-06 14:30:00	1
The Matrix Reloaded	18	2024-02-06 17:00:00	1
Eternal Sunshine of the Spotless Mind	19	2024-02-06 19:30:00	1
The Lion King	20	2024-02-06 15:30:00	1
Gladiator	21	2024-02-06 18:15:00	1
Inglourious Basterds	22	2024-02-06 16:45:00	1
Back to the Future	23	2024-02-06 19:45:00	1
The Godfather: Part II	24	2024-02-06 14:15:00	1
The Avengers	25	2024-02-06 17:45:00	1