MovieMetrics

A movie booking system analysis

By : Aditi Shetti

Problem Statement:

An online movie booking platform is facing challenges due to limited Insights on resource allocation, user preferences and optimizing movie scheduling .

There is a need for a data analysis project to gain insights into audience preferences, user engagement , genre popularity and theatre performance and booking behaviors.

Business Context:

This data analysis project on movie booking platform analysis serves to provide insights to stakeholders, management, marketing teams, etc within the organisation.

Understanding user demographics, genre popularity , theatre performance , language diversity ,etc to help make Data-Driven decisions , enhanced customer experience , sales and revenue optimisation and improve overall operations.

Project Overview:

The primary objective of MovieMetrics is to develop a movie booking system that tracks and analyzes movie screenings, user bookings, and theatre performance.

The database includes tables for movies, theatres, shows, users, and bookings, allowing for in-depth insights into ticket sales, user demographics, and movie performance.

Through various SQL queries, I explored relationships and trends, facilitating better decision-making for cinema operations.

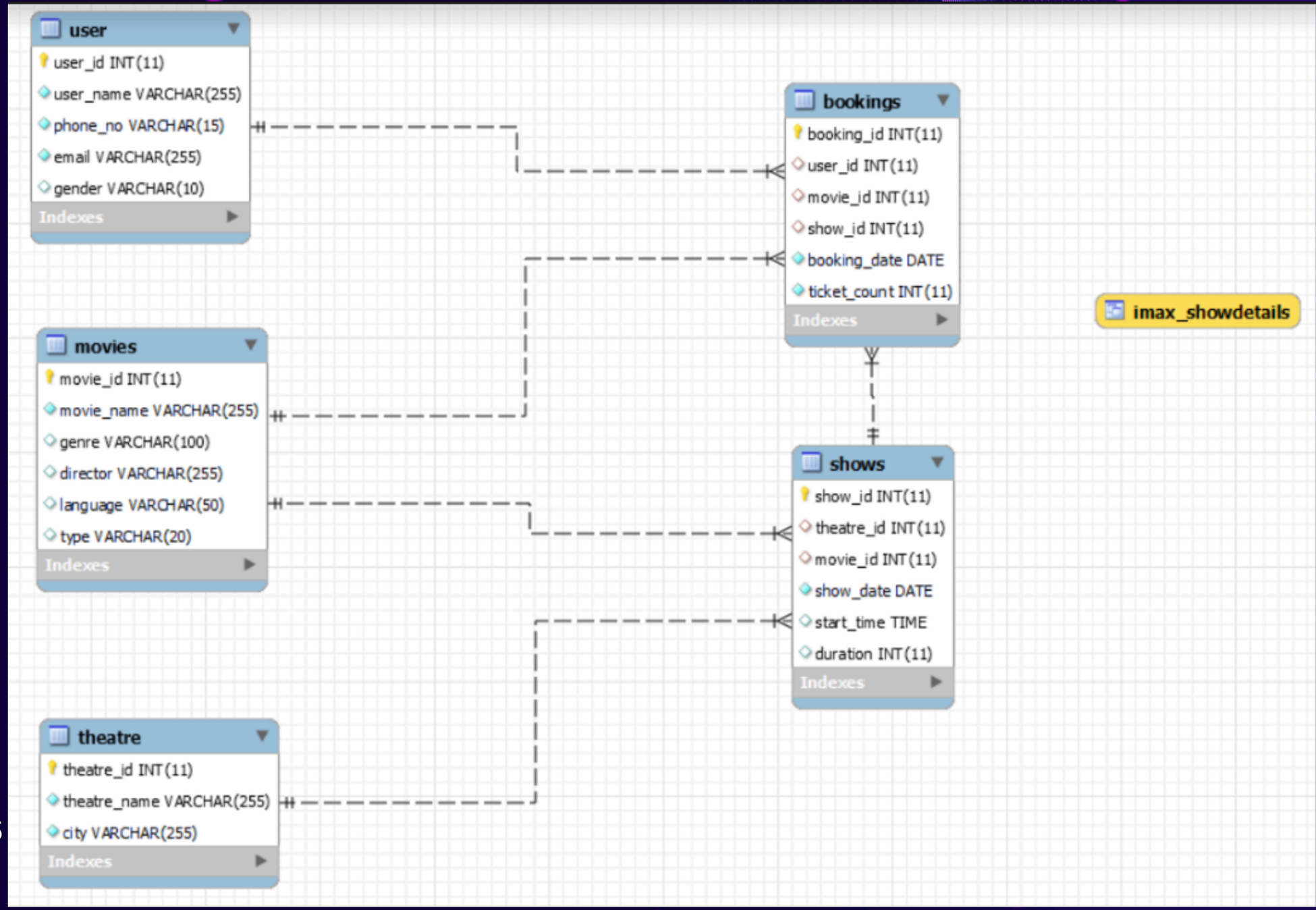
Database Schema

Tables:

- user
- movies
- theatre
- shows
- bookings

VIEWS:

Imax_showdetails



OVERVIEW OF ALL TABLES

select * from movies

movie_id	movie_name	genre	director	language	type
1001	Inception	Sci-Fi	Christopher Nolan	English	IMAX
1002	Dil Chahta Hai	Drama	Farhan Akhtar	Hindi	2D
1003	Baahubali: The Beginning	Action	S. S. Rajamouli	Telugu	IMAX
1004	Spirited Away	Animation	Hayao Miyazaki	Japanese	2D
1005	Zindagi Na Milegi Dobara	Comedy	Zoya Akhtar	Hindi	2D
1006	The Shawshank Redemption	Drama	Frank Darabont	English	2D
1007	Lagaan	Historical	Ashutosh Gowariker	Hindi	2D
1008	Dangal	Sports	Nitesh Tiwari	Hindi	2D
1009	The Dark Knight	Action	Christopher Nolan	English	IMAX
1010	Drishyam	Thriller	Jeethu Joseph	Malayalam	2D
1011	Kantara	Action	Rishab Shetty	Kannada	IMAX
1012	The Handmaiden	Thriller	Park Chan-wook	Korean	2D
1013	Your Name	Animation	Makoto Shinkai	Japanese	2D
1014	Swaas	Drama	Madhur Bhandarkar	Marathi	2D
1015	Mangal Pandey: The Rising	Historical	Ketan Mehta	Hindi	IMAX
1016	Kahaani	Thriller	Sujoy Ghosh	Hindi	2D
1017	Avatar	Sci-Fi	James Cameron	English	3D
1018	PK	Comedy	Rajkumar Hirani	Hindi	IMAX
1019	The Lion King	Animation	Jon Favreau	English	IMAX
1020	Gully Boy	Drama	Zoya Akhtar	Hindi	IMAX

select * from theatre

theatre_id	theatre_name	city
2001	IMAX Cinemas	Mumbai
2002	PVR Cinemas	Delhi
2003	Cinepolis	Bangalore
2004	Inox	Hyderabad
2005	Fun Cinemas	Chennai
2006	Carnival Cinemas	Kolkata
2007	Satyam Cineplexes	Jaipur
2008	Silver Screens	Pune
2009	Big Cinemas	Ahmedabad
2010	Cineworld	Mumbai

select * from bookings

booking_id	user_id	movie_id	show_id	booking_date	ticket_count
5001	3001	1005	4001	2024-01-15	2
5002	3002	1003	4002	2024-02-20	3
5003	3003	1007	4003	2024-03-05	4
5004	3004	1010	4004	2024-03-25	1
5005	3005	1017	4005	2024-04-10	5
5006	3006	1002	4006	2024-04-20	2
5007	3007	1015	4007	2024-05-05	3
5008	3008	1009	4008	2024-05-15	2
5009	3009	1012	4009	2024-06-01	1
5010	3010	1018	4010	2024-06-15	4
5011	3001	1019	4011	2024-06-28	3
5012	3005	1008	4012	2024-07-05	4

select * from user

user_id	user_name	phone_no	email	gender
3001	Aditi Shetti	9876543210	aditi.shetti@example.com	Female
3002	Priya Patel	9123456789	priya.patel@example.com	Female
3003	Ravi Kumar	9988776655	ravi.kumar@example.com	Male
3004	Sneha Gupta	9898989898	sneha.gupta@example.com	Female
3005	Rahul Mehta	9765432109	rahul.mehta@example.com	Male
3006	Aarti Shetti	9654321098	aarti.shetti@example.com	Female
3007	Vikram Singh	9556677889	vikram.singh@example.com	Male
3008	Neha Reddy	9445566778	neha.reddy@example.com	Male
3009	Diya Kasbekar	9222333444	diya.kasbekar@example.com	Female
3010	Rohan Joshi	9333221111	rohan.joshi@example.com	Male

select * from shows

show_id	theatre_id	movie_id	show_date	start_time	duration
4001	2001	1005	2024-01-19	18:30:00	155
4002	2002	1003	2024-02-21	20:00:00	180
4003	2003	1007	2024-03-07	14:00:00	210
4004	2004	1010	2024-03-29	16:00:00	160
4005	2005	1017	2024-04-15	19:30:00	162
4006	2006	1002	2024-05-27	21:00:00	180
4007	2007	1015	2024-05-10	12:00:00	180
4008	2008	1009	2024-05-19	11:00:00	152
4009	2009	1012	2024-06-07	22:00:00	145
4010	2010	1018	2024-06-16	15:30:00	155
4011	2001	1019	2024-07-01	17:00:00	120
4012	2003	1008	2024-07-10	19:30:00	140
4013	2004	1011	2024-07-15	15:00:00	150

1. List all movies with their genres and directors.

Query:

```
select genre,director from movies
```

Output:

genre	director
Sci-Fi	Christopher Nolan
Drama	Farhan Akhtar
Action	S. S. Rajamouli
Animation	Hayao Miyazaki
Comedy	Zoya Akhtar
Drama	Frank Darabont
Historical	Ashutosh Gowariker
Sports	Nitesh Tiwari
Action	Christopher Nolan
Thriller	Jeethu Joseph
Action	Rishab Shetty

2. List all directors according to their movies genres

Query:

```
select genre, group_concat(distinct director order by director)as directors_genre
from movies group by genre
```

Output:

genre	directors_genre
Action	Christopher Nolan,Rishab Shetty,S. S. Rajamouli
Animation	Hayao Miyazaki,Jon Favreau,Makoto Shinkai
Comedy	Rajkumar Hirani,Zoya Akhtar
Drama	Farhan Akhtar,Frank Darabont,Madhur Bhandarkar,Zoya Akhtar
Historical	Ashutosh Gowariker,Ketan Mehta
Sci-Fi	Christopher Nolan,James Cameron
Sports	Nitesh Tiwari
Thriller	Jeethu Joseph,Park Chan-wook,Sujoy Ghosh

3. How many unique languages are present, and what are they?

Query:

```
select count(distinct language)as count_lang from movies
select distinct language as distinct_lang from movies
```

Output:

count_lang
8

distinct_lang
English
Hindi
Telugu
Japanese
Malayalam
Kannada
Korean
Marathi

4. Show movies according to different format types.

Query:

```
select type,
group_concat(distinct movie_name order by movie_name)as movie_type
from movies group by type
```

Output:

type	movie_type
2D	Dangal,Dil Chahta Hai,Drishyam,Kahaani,Lagaan,Spirited Away,Swaas,The Handmaiden,The Shawshank Redemption,Your Name,Zindagi Na Milegi Dobara
3D	Avatar
IMAX	Baahubali: The Beginning,Gully Boy,Inception,Kantara,Mangal Pandey: The Rising,PK,The Dark Knight,The Lion King

5. Display movies according to languages.

Query:

```
select movie_id,movie_name,language
from movies order by language
```

Output:

movie_id	movie_name	language
1001	Inception	English
1006	The Shawshank Redemption	English
1009	The Dark Knight	English
1017	Avatar	English
1019	The Lion King	English
1002	Dil Chahta Hai	Hindi
1005	Zindagi Na Milegi Dobara	Hindi
1007	Lagaan	Hindi
1008	Dangal	Hindi
1015	Mangal Pandey: The Rising	Hindi
1016	Kahaani	Hindi
1018	PK	Hindi
1020	Gully Boy	Hindi
1004	Spirited Away	Japanese
1013	Your Name	Japanese
1011	Kantara	Kannada

6. Find all theatres located in Mumbai

Query:

```
select * from theatre
where city = 'Mumbai';
```

Output:

theatre_id	theatre_name	city
2001	IMAX Cinemas	Mumbai
2010	Cineworld	Mumbai

7. Retrieve the names and email addresses of all female users

Query:

```
select user_name , email
from user where gender = 'Female'
```

Output:

user_name	email
Aditi Shetti	aditi.shetti@example.com
Priya Patel	priya.patel@example.com
Sneha Gupta	sneha.gupta@example.com
Aarti Shetti	aarti.shetti@example.com
Diya Kasbekar	diya.kasbekar@example.com

8. List the names and durations of movies longer than 120 minutes.

Query:

```
select m.movie_name , s.duration
from movies as m
join shows as s
on m.movie_id = s.movie_id
where s.duration > 120
```

Output:

movie_name	duration
Zindagi Na Milegi Dobara	155
Baahubali: The Beginning	180
Lagaan	210
Drishyam	160
Avatar	162
Dil Chahta Hai	180
Mangal Pandey: The Rising	180
The Dark Knight	152
The Handmaiden	145
PK	155
Dangal	140
Kantara	150

9. Get movie names that are in the 'Drama', 'Comedy', or 'Thriller' genre.

Query:

```
select movie_id, movie_name
from movies where genre in ('Drama', 'Comedy','Thriller')
```

Output:

movie_id	movie_name
1002	Dil Chahta Hai
1005	Zindagi Na Milegi Dobara
1006	The Shawshank Redemption
1010	Drishyam
1012	The Handmaiden
1014	Swaas
1016	Kahaani
1018	PK
1020	Gully Boy

10.total tickets sold

Query:

```
select sum(ticket_count) from bookings
```

Output:

sum(ticket_count)
34

11. Find the average duration of movies in the dataset.

Query:

```
select round(avg(duration),2)as avg_movie_duration
from shows
```

Output:

avg_movie_duration
160.69

12. Find the movie with longest runtime

Query:

```
select m.movie_name , s.duration
from shows as s
join movies as m
on m.movie_id = s.movie_id
order by duration desc limit 1
```

Output:

movie_name	duration
Lagaan	210

13. All Shows in the month of March to May

Query:

```
select * from shows
where show_date between '2024-03-01' and '2024-05-31'
```

Output:

show_id	theatre_id	movie_id	show_date	start_time	duration
4003	2003	1007	2024-03-07	14:00:00	210
4004	2004	1010	2024-03-29	16:00:00	160
4005	2005	1017	2024-04-15	19:30:00	162
4006	2006	1002	2024-05-27	21:00:00	180
4007	2007	1015	2024-05-10	12:00:00	180
4008	2008	1009	2024-05-19	11:00:00	152

14. Names of all movies which users have booked

Query:

```
select m.movie_name,
       m.movie_name,
       b.booking_id,
       b.booking_date,
       b.ticket_count
from movies as m
join bookings as b
on m.movie_id = b.movie_id
```

Output:

movie_name	movie_name	booking_id	booking_date	ticket_count
Zindagi Na Milegi Dobara	Zindagi Na Milegi Dobara	5001	2024-01-15	2
Baahubali: The Beginning	Baahubali: The Beginning	5002	2024-02-20	3
Lagaan	Lagaan	5003	2024-03-05	4
Drishyam	Drishyam	5004	2024-03-25	1
Avatar	Avatar	5005	2024-04-10	5
Dil Chahta Hai	Dil Chahta Hai	5006	2024-04-20	2
Mangal Pandey: The Rising	Mangal Pandey: The Rising	5007	2024-05-05	3
The Dark Knight	The Dark Knight	5008	2024-05-15	2
The Handmaiden	The Handmaiden	5009	2024-06-01	1
PK	PK	5010	2024-06-15	4
The Lion King	The Lion King	5011	2024-06-28	3
Dangal	Dangal	5012	2024-07-05	4

15. Find all users who have booked tickets for the movie "Avatar".

Query:

```
select u.user_name, b.booking_id,b.user_id, b.ticket_count ,m.movie_name
from bookings as b
join movies as m on m.movie_id = b.movie_id
join user as u  on u.user_id = b.user_id
where m.movie_name = 'Avatar'
```

Output:

user_name	booking_id	user_id	ticket_count	movie_name
Rahul Mehta	5005	3005	5	Avatar

16. Retrieve theatre names where the theatre names do not contain 'Cinemas'

Query:

```
select theatre_name, city
from theatre
where theatre_name not like '%Cinemas%'
```

Output:

theatre_name	city
Cinepolis	Bangalore
Inox	Hyderabad
Satyam Cineplexes	Jaipur
Silver Screens	Pune
Cineworld	Mumbai

17. Get the names of all movies playing in Mumbai or Pune or Hyderabad.

Query:

```
select s.show_date, m.movie_name , t.theatre_name,t.city
from theatre t
join shows s on  s.theatre_id= t.theatre_id
join movies m on m.movie_id = s.movie_id
where t.city in ('Mumbai','Pune','Hyderabad')
```

Output:

show_date	movie_name	theatre_name	city
2024-01-19	Zindagi Na Milegi Dobara	IMAX Cinemas	Mumbai
2024-03-29	Drishyam	Inox	Hyderabad
2024-05-19	The Dark Knight	Silver Screens	Pune
2024-06-16	PK	Cineworld	Mumbai
2024-07-01	The Lion King	IMAX Cinemas	Mumbai
2024-07-15	Kantara	Inox	Hyderabad

18. Find movie names and theatre details where the theatre is ‘PVR Cinemas'and the city is 'Delhi'

Query:

```
select m.movie_name , t.theatre_name,t.city
from theatre t
join shows s on s.theatre_id= t.theatre_id
join movies m on m.movie_id = s.movie_id
where t.city = 'Delhi' and t.theatre_name= 'PVR Cinemas'
```

Output:

movie_name	theatre_name	city
Baahubali: The Beginning	PVR Cinemas	Delhi

19. Movies playing in theatres except Cinepolis

Query:

```
select m.movie_name , t.theatre_name,t.city
from theatre t
join shows s on  s.theatre_id= t.theatre_id
join movies m  on m.movie_id = s.movie_id
where t.theatre_name not like  'Cinepolis'
```

Output:

movie_name	theatre_name	city
Zindagi Na Milegi Dobara	IMAX Cinemas	Mumbai
Baahubali: The Beginning	PVR Cinemas	Delhi
Drishyam	Inox	Hyderabad
Avatar	Fun Cinemas	Chennai
Dil Chahta Hai	Carnival Cinemas	Kolkata
Mangal Pandey: The Rising	Satyam Cineplexes	Jaipur
The Dark Knight	Silver Screens	Pune
The Handmaiden	Big Cinemas	Ahmedabad
PK	Cineworld	Mumbai
The Lion King	IMAX Cinemas	Mumbai
Kantara	Inox	Hyderabad

20. Find the total number of tickets booked for each movie.

Query:

```
select m.movie_id,m.movie_name, sum(b.ticket_count)as total_tickets
from bookings b
join movies m on m.movie_id = b.movie_id
group by m.movie_name
```

Output:

movie_id	movie_name	total_tickets
1017	Avatar	5
1003	Baahubali: The Beginning	3
1008	Dangal	4
1002	Dil Chahta Hai	2
1010	Drishyam	1
1007	Lagaan	4
1015	Mangal Pandey: The Rising	3
1018	PK	4
1009	The Dark Knight	2
1012	The Handmaiden	1
1019	The Lion King	3
1005	Zindagi Na Milegi Dobara	2

21. Retrieve the details of the most popular movie (the movie with the highest tickets sold).

Query:

```
select m.movie_id,m.movie_name, sum(b.ticket_count) as total_tickets
from bookings b
join movies m on m.movie_id = b.movie_id
group by m.movie_name
order by b.ticket_count desc limit 1
```

Output:

movie_id	movie_name	total_tickets
1017	Avatar	5

22. Get the total number of bookings made by each user.

Query:

```
select u.user_name, count(booking_id) as total_bookings
from bookings b
join user u on u.user_id = b.user_id
group by u.user_name order by total_bookings desc
```

Output:

user_name	total_bookings
Aditi Shetti	2
Rahul Mehta	2
Priya Patel	1
Ravi Kumar	1
Sneha Gupta	1
Aarti Shetti	1
Vikram Singh	1
Neha Reddy	1
Diya Kasbekar	1
Rohan Joshi	1

23. Find users who have booked more than 2 tickets for any show.

Query:

```
select u.user_id , u.user_name from bookings b
join user u on u.user_id = b.user_id
 where ticket_count>2
```

Output:

user_id	user_name
3002	Priya Patel
3003	Ravi Kumar
3005	Rahul Mehta
3007	Vikram Singh
3010	Rohan Joshi
3001	Aditi Shetti
3005	Rahul Mehta

24. List all movies that have not been booked by any user.

Query:

```
select *
from movies m
left join bookings b on m.movie_id= b.movie_id
where b.movie_id is null
```

Output:

movie_id	movie_name	genre	director	language	type	booking_id	user_id	movie_id	show_id	booking_date	ticket_count
1001	Inception	Sci-Fi	Christopher Nolan	English	IMAX	NULL	NULL	NULL	NULL	NULL	NULL
1004	Spirited Away	Animation	Hayao Miyazaki	Japanese	2D	NULL	NULL	NULL	NULL	NULL	NULL
1006	The Shawshank Redemption	Drama	Frank Darabont	English	2D	NULL	NULL	NULL	NULL	NULL	NULL
1011	Kantara	Action	Rishab Shetty	Kannada	IMAX	NULL	NULL	NULL	NULL	NULL	NULL
1013	Your Name	Animation	Makoto Shinkai	Japanese	2D	NULL	NULL	NULL	NULL	NULL	NULL
1014	Swaas	Drama	Madhur Bhandarkar	Marathi	2D	NULL	NULL	NULL	NULL	NULL	NULL
1016	Kahaani	Thriller	Sujoy Ghosh	Hindi	2D	NULL	NULL	NULL	NULL	NULL	NULL
1020	Gully Boy	Drama	Zoya Akhtar	Hindi	IMAX	NULL	NULL	NULL	NULL	NULL	NULL

25. Retrieve name of the month from the dates.

Query:

```
select distinct monthname(booking_date)
from bookings
```

Output:

monthname(booking_date)
January
February
March
April
May
June
July

26. Find all the bookings made in the month of May and June.

Query:

```
select user_id , booking_date ,ticket_count
from bookings where monthname(booking_date) in ('May','June')
```

Output:

user_id	booking_date	ticket_count
3007	2024-05-05	3
3008	2024-05-15	2
3009	2024-06-01	1
3010	2024-06-15	4
3001	2024-06-28	3

27. Change format of the booking date

Query:

```
select booking_id, date_format(booking_date,'%d-%m-%Y')as booking_date_dmy
from bookings
```

Output:

booking_id	booking_date_dmy
5001	15-01-2024
5002	20-02-2024
5003	05-03-2024
5004	25-03-2024
5005	10-04-2024
5006	20-04-2024
5007	05-05-2024
5008	15-05-2024
5009	01-06-2024

28. Find the difference between the booking date and actual show date

Query:

```
select m.movie_name,datediff(s.show_date,b.booking_date) as days_for_show
from bookings b
join shows s on  b.show_id= s.show_id
join movies m on m.movie_id = b.movie_id
```

Output:

movie_name	days_for_show
Zindagi Na Milegi Dobara	4
Baahubali: The Beginning	1
Lagaan	2
Drishyam	4
Avatar	5
Dil Chahta Hai	37
Mangal Pandey: The Rising	5
The Dark Knight	4
The Handmaiden	6
PK	1
The Lion King	3
Dangal	5

29. Change values of show_date to a month later

Query:

```
select show_date,
       date_add( show_date, interval 1 month)as show_date_add_1m
from shows
```

Output:

show_date	show_date_add_1m
2024-01-19	2024-02-19
2024-02-21	2024-03-21
2024-03-07	2024-04-07
2024-03-29	2024-04-29
2024-04-15	2024-05-15
2024-05-27	2024-06-27
2024-05-10	2024-06-10
2024-05-19	2024-06-19
2024-06-07	2024-07-07

30. Change values of booking_date to 5 days earlier

Query:

```
select booking_date,date_sub(booking_date, interval 5 day)as booking_date_sub_5d
from bookings
```

Output:

booking_date	booking_date_sub_5d
2024-01-15	2024-01-10
2024-02-20	2024-02-15
2024-03-05	2024-02-29
2024-03-25	2024-03-20
2024-04-10	2024-04-05
2024-04-20	2024-04-15
2024-05-05	2024-04-30

31. What is the most frequent day of the week for movie shows?

Query:

```
select show_date,dayname(show_date)as show_date_dayname, count(*) as show_count
from shows group by show_date_dayname order by show_count desc
```

Output:

show_date	show_date_dayname	show_count
2024-01-19	Friday	4
2024-04-15	Monday	4
2024-05-19	Sunday	2
2024-02-21	Wednesday	2
2024-03-07	Thursday	1

32. What day of the week corresponds to each booking date in the dataset?

Query:

```
select booking_date,dayofweek(booking_date)as booking_date_dayofweek
from bookings
```

Output:

booking_date	booking_date_dayofweek
2024-01-15	2
2024-02-20	3
2024-03-05	3
2024-03-25	2
2024-04-10	4
2024-04-20	7

33. What is the day of the year for each booking date in the dataset?

Query:

```
select booking_date, dayofyear(booking_date) as booking_date_dayofyear
from bookings
```

Output:

booking_date	booking_date_dayofyear
2024-01-15	15
2024-02-20	51
2024-03-05	65
2024-03-25	85
2024-04-10	101
2024-04-20	111
2024-05-05	126
2024-05-15	136
2024-06-01	153
2024-06-15	167
2024-06-28	180
2024-07-05	187

34. What is the distribution of movie showtimes throughout the day?

Query:

```
select start_time, extract(hour from start_time) as start_time_hour
from shows
```

Output:

start_time	start_time_hour
18:30:00	18
20:00:00	20
14:00:00	14
16:00:00	16
19:30:00	19
21:00:00	21
12:00:00	12
11:00:00	11
22:00:00	22
15:30:00	15
17:00:00	17
19:30:00	19
15:00:00	15

35. Get a list of the top 3 movies with the most tickets booked, but skip the most booked movie.

Query:

```
select u.user_id , u.user_name , m.movie_name ,b.ticket_count
from bookings b
join user u on u.user_id = b.user_id
join movies m on b.movie_id = m.movie_id
order by b.ticket_count desc
limit 3 offset 1
```

Output:

user_id	user_name	movie_name	ticket_count
3005	Rahul Mehta	Dangal	4
3003	Ravi Kumar	Lagaan	4
3010	Rohan Joshi	PK	4

36. Find the average ticket count per booking for each user.

Query:

```
select u.user_id , u.user_name, round(avg(b.ticket_count)) as avg_ticket_count
from user u
join bookings b
on b.user_id= u.user_id
group by u.user_id, u.user_name
order by avg_ticket_count desc
```

Output:

user_id	user_name	avg_ticket_count
3005	Rahul Mehta	5
3010	Rohan Joshi	4
3003	Ravi Kumar	4
3007	Vikram Singh	3
3001	Aditi Shetti	3
3002	Priya Patel	3
3006	Aarti Shetti	2
3008	Neha Reddy	2
3004	Sneha Gupta	1
3009	Diya Kasbekar	1

37. Which booking had the highest number of tickets sold?

Query:

```
select t.theatre_name, s.start_time, s.show_date, b.ticket_count
from bookings b
join shows s on b.show_id = s.show_id
join theatre t on s.theatre_id = t.theatre_id
where b.ticket_count = (
    select MAX(b2.ticket_count)
    from bookings b2 )
limit 1;
```

Output:

theatre_name	start_time	show_date	ticket_count
Fun Cinemas	19:30:00	2024-04-15	5

38. List all users who have booked tickets for shows scheduled after '2024-04-01' but for less than 5 tickets.

Query:

```
select u.user_id, u.user_name , u.email , s.show_date , b.ticket_count
from bookings b
join user u on b.user_id = u.user_id
join shows s on b.show_id = s.show_id
where s.show_date > '2024-04-01' and b.ticket_count < 5
```

Output:

user_id	user_name	email	show_date	ticket_count
3006	Aarti Shetti	aarti.shetti@example.com	2024-05-27	2
3007	Vikram Singh	vikram.singh@example.com	2024-05-10	3
3008	Neha Reddy	neha.reddy@example.com	2024-05-19	2
3009	Diya Kasbekar	diya.kasbekar@example.com	2024-06-07	1
3010	Rohan Joshi	rohan.joshi@example.com	2024-06-16	4
3001	Aditi Shetti	aditi.shetti@example.com	2024-07-01	3
3005	Rahul Mehta	rahul.mehta@example.com	2024-07-10	4

39. Retrieve the users who have never booked any tickets for movies in the 'Comedy' genre.

Query:

```
select u.user_id, u.user_name
from user u
where not exists (
  select 1
  from bookings b
  join movies m on b.movie_id = m.movie_id
  where  b.user_id = u.user_id and m.genre = 'Comedy')
```

Output:

user_id	user_name
3002	Priya Patel
3003	Ravi Kumar
3004	Sneha Gupta
3005	Rahul Mehta
3006	Aarti Shetti
3007	Vikram Singh
3008	Neha Reddy
3009	Diya Kasbekar

40. Find the average duration of shows for each theatre.

Query:

```
select t.theatre_id,t.theatre_name,
       round(avg(s.duration),0) as avg_duration
from shows s
join theatre t
on t.theatre_id= s.theatre_id
group by t.theatre_id
order by avg_duration desc
```

Output:

theatre_id	theatre_name	avg_duration
2007	Satyam Cineplexes	180
2002	PVR Cinemas	180
2006	Carnival Cinemas	180
2003	Cinepolis	175
2005	Fun Cinemas	162
2010	Cineworld	155
2004	Inox	155
2008	Silver Screens	152
2009	Big Cinemas	145
2001	IMAX Cinemas	138

41. Rank Users by Ticket Count in all cities.

Query:

```
Select t.city ,u.user_name, b.ticket_count,
   rank() over (partition by t.city order by b.ticket_count desc)as rank_of_ticket_in_cities
from bookings b
join user u on b.user_id = u.user_id
join shows s on s.show_id = b.show_id
join theatre t on s.theatre_id = t.theatre_id
```

Output:

city	user_name	ticket_count	rank_of_ticket_in_cities
Ahmedabad	Diya Kasbekar	1	1
Bangalore	Ravi Kumar	4	1
Bangalore	Rahul Mehta	4	1
Chennai	Rahul Mehta	5	1
Delhi	Priya Patel	3	1
Hyderabad	Sneha Gupta	1	1
Jaipur	Vikram Singh	3	1
Kolkata	Aarti Shetti	2	1
Mumbai	Rohan Joshi	4	1
Mumbai	Aditi Shetti	3	2
Mumbai	Aditi Shetti	2	3
Pune	Neha Reddy	2	1

42. Rank Users by Ticket Count Across All Theatres

Query:

```
Select t.city ,u.user_name, b.ticket_count, t.theatre_name,
   rank() over (partition by t.theatre_name order by b.ticket_count desc) as rank_tcount_theatre
from bookings b
join user u on b.user_id = u.user_id
join shows s on s.show_id = b.show_id
join theatre t on s.theatre_id = t.theatre_id
```

Output:

city	user_name	ticket_count	theatre_name	rank_tcount_theatre
Ahmedabad	Diya Kasbekar	1	Big Cinemas	1
Kolkata	Aarti Shetti	2	Carnival Cinemas	1
Bangalore	Ravi Kumar	4	Cinepolis	1
Bangalore	Rahul Mehta	4	Cinepolis	1
Mumbai	Rohan Joshi	4	Cineworld	1
Chennai	Rahul Mehta	5	Fun Cinemas	1
Mumbai	Aditi Shetti	3	IMAX Cinemas	1
Mumbai	Aditi Shetti	2	IMAX Cinemas	2
Hyderabad	Sneha Gupta	1	Inox	1
Delhi	Priya Patel	3	PVR Cinemas	1
Jaipur	Vikram Singh	3	Satyam Cineplexes	1
Pune	Neha Reddy	2	Silver Screens	1

43. Rank tickets across cities according to gender

Query:

```
Select  u.gender, t.city ,u.user_name, b.ticket_count,
   dense_rank() over (partition by u.gender order by b.ticket_count desc) as rank_tcount_gender
from bookings b
join user u on b.user_id = u.user_id
join shows s on b.show_id= s.show_id
join theatre t on t.theatre_id = s.theatre_id
```

Output:

gender	city	user_name	ticket_count	rank_tcount_gender
Female	Mumbai	Aditi Shetti	3	1
Female	Delhi	Priya Patel	3	1
Female	Mumbai	Aditi Shetti	2	2
Female	Kolkata	Aarti Shetti	2	2
Female	Ahmedabad	Diya Kasbekar	1	3
Female	Hyderabad	Sneha Gupta	1	3
Male	Chennai	Rahul Mehta	5	1
Male	Mumbai	Rohan Joshi	4	2
Male	Bangalore	Rahul Mehta	4	2
Male	Bangalore	Ravi Kumar	4	2
Male	Jaipur	Vikram Singh	3	3
Male	Pune	Neha Reddy	2	4

44. Ranking Users by Movie in Each Genre

Query:

```
select u.user_id, u.user_name, m.genre,
  dense_rank() over (order by m.genre) as rank_user_genre
from user u
join bookings b on b.user_id= u.user_id
join movies m on m.movie_id= b.movie_id
```

Output:

user_id	user_name	genre	rank_user_genre
3008	Neha Reddy	Action	1
3002	Priya Patel	Action	1
3001	Aditi Shetti	Animation	2
3001	Aditi Shetti	Comedy	3
3010	Rohan Joshi	Comedy	3
3006	Aarti Shetti	Drama	4
3007	Vikram Singh	Historical	5
3003	Ravi Kumar	Historical	5
3005	Rahul Mehta	Sci-Fi	6
3005	Rahul Mehta	Sports	7
3004	Sneha Gupta	Thriller	8
3009	Diya Kasbekar	Thriller	8

45. Write a query to swap genders

Query:

```
update user set gender= case
    when gender = 'Female' then  'Male'
    when gender = 'Male' then  'Female'
end
```

Output:

user_name	gender
Aditi Shetti	Male
Priya Patel	Male
Ravi Kumar	Female
Sneha Gupta	Male
Rahul Mehta	Female
Aarti Shetti	Male
Vikram Singh	Female
Neha Reddy	Female
Diya Kasbekar	Male
Rohan Joshi	Female

46. Create a View where movie type is IMAX

Query:

```
create view IMAX_ShowDetails as (
select m.movie_name ,t.theatre_name ,t.city,s.show_date ,s.start_time
from theatre t
join shows s on t.theatre_id = s.theatre_id
join movies m on m.movie_id = s.movie_id
where m.type = 'IMAX'
)
```

Output:

movie_name	theatre_name	city	show_date	start_time
Baahubali: The Beginning	PVR Cinemas	Delhi	2024-02-21	20:00:00
Mangal Pandey: The Rising	Satyam Cineplexes	Jaipur	2024-05-10	12:00:00
The Dark Knight	Silver Screens	Pune	2024-05-19	11:00:00
PK	Cineworld	Mumbai	2024-06-16	15:30:00
The Lion King	IMAX Cinemas	Mumbai	2024-07-01	17:00:00
Kantara	Inox	Hyderabad	2024-07-15	15:00:00

47. A Stored procedure to Get Theatre Details by City

Query:

```
DELIMITER //
create procedure TheatreDetailsByCity(IN CITYNAME varchar(100))
begin
  select t.theatre_name, t.city
  from theatre t
  where t.city = CITYNAME;
end //

call TheatreDetailsByCity('pune')
call TheatreDetailsByCity('kolkata')
```

Output:

theatre_name	city
Silver Screens	Pune

theatre_name	city
Carnival Cinemas	Kolkata

48. A Stored Procedure Get all User Booking Details

Query:

```
DELIMITER //
create procedure GetUserBookingDetails(IN USERID INT )
begin
  select u.user_name,
         m.movie_name,
         m.type ,
         s.show_date,
         t.theatre_name,
         b.ticket_count
  from user u
  join bookings b on u.user_id= b.user_id
  join movies  m  on b.movie_id= m.movie_id
  join shows  s   on b.show_id= s.show_id
  join theatre t on s.theatre_id = t.theatre_id
  where u.user_id = USERID;
end //
```

```
call GetUserBookingDetails(3001)

call GetUserBookingDetails(3005)
```

Output:

user_name	movie_name	type	show_date	theatre_name	ticket_count
Aditi Shetti	Zindagi Na Milegi Dobara	2D	2024-01-19	IMAX Cinemas	2
Aditi Shetti	The Lion King	IMAX	2024-07-01	IMAX Cinemas	3

user_name	movie_name	type	show_date	theatre_name	ticket_count
Rahul Mehta	Avatar	3D	2024-04-15	Fun Cinemas	5
Rahul Mehta	Dangal	2D	2024-07-10	Cinepolis	4

Key Findings

- The longest movie in the dataset is Lagaan, which has a duration of 210 minutes and falls under the 'Historical' genre.
- The average duration of the movies is 160.69 minutes.
- Found that users are more likely to book a movie within one week before the show's scheduled date.
- Identified users who have registered but haven't made any bookings.
- Calculated the average duration of shows across different theatres.
- Created a view to filter and display only IMAX-type movies.
- Developed a stored procedure to retrieve theatre details based on city.
- Created a stored procedure to fetch detailed user booking information.

Thank You

By : Aditi Shetti