

```
mysql> /* ASSIGNMENT
/*> TASK 1 :DATABASE DEISGN :
/*> 1. Create the database named "TicketBookingSystem"
/*> */
```

```
mysql> CREATE DATABASE TicketBookingSystem;
Query OK, 1 row affected (0.00 sec)
```

```
mysql> USE TicketBookingSystem;
Database changed
```

```
mysql> /* 2. Write SQL scripts to create the mentioned tables with appropriate data types, constraints, and relationships. -VENUE -EVENT -CUSTOMER -BOOKING*/
```

```
mysql> /*VENUE*/
```

```
mysql> CREATE TABLE Venue (venue_id INT PRIMARY KEY, venue_name TEXT, address TEXT);
Query OK, 0 rows affected (0.02 sec)
```

```
mysql> desc Venue;
```

Field	Type	Null	Key	Default	Extra
venue_id	int	NO	PRI	NULL	
venue_name	text	YES		NULL	
address	text	YES		NULL	

3 rows in set (0.00 sec)

```
mysql> /*Events*/
```

```
mysql> CREATE TABLE Event (event_id INT PRIMARY KEY, event_name TEXT, event_date DATE, event_time TIME, venue_id INT,
total_seats INT, available_seats INT, ticket_price DECIMAL, event_type ENUM('Movie', 'Sports', 'Concert'),
booking_id INT, FOREIGN KEY (venue_id) REFERENCES Venue(venue_id));
Query OK, 0 rows affected (0.04 sec)
```

```
mysql> desc Event;
```

Field	Type	Null	Key	Default	Extra
event_id	int	NO	PRI	NULL	
event_name	text	YES		NULL	
event_date	date	YES		NULL	
event_time	time	YES		NULL	
venue_id	int	YES	MUL	NULL	
total_seats	int	YES		NULL	
available_seats	int	YES		NULL	
ticket_price	decimal(10,0)	YES		NULL	
event_type	enum('Movie','Sports','Concert')	YES		NULL	

Activate Windows
Go to Settings to activate Windows.



Type here to search



22°C



ENG

20:15
12-12-2023



ticket_price	decimal(10,0)	YES		NULL	
event_type	enum('Movie','Sports','Concert')	YES		NULL	
booking_id	int	YES		NULL	

10 rows in set (0.00 sec)

mysql> /*Booking*/

```
mysql> CREATE TABLE Booking (booking_id INT PRIMARY KEY, customer_id INT, event_id INT, num_tickets INT,
                             total_cost DECIMAL, booking_date DATE, FOREIGN KEY (event_id) REFERENCES Event(event_id));
```

Query OK, 0 rows affected (0.03 sec)

mysql> /*Customer*/

```
mysql> CREATE TABLE Customer (customer_id INT PRIMARY KEY, customer_name TEXT, email TEXT,
                             phone_number TEXT, booking_id INT, FOREIGN KEY (booking_id) REFERENCES Booking(booking_id));
```

Query OK, 0 rows affected (0.03 sec)

mysql> desc Customer;

Field	Type	Null	Key	Default	Extra
customer_id	int	NO	PRI	NULL	
customer_name	text	YES		NULL	
email	text	YES		NULL	
phone_number	text	YES		NULL	
booking_id	int	YES	MUL	NULL	

5 rows in set (0.00 sec)

mysql> ALTER TABLE Event

-> ADD CONSTRAINT fk_booking

-> FOREIGN KEY (booking_id)

-> REFERENCES Booking(booking_id);

Query OK, 0 rows affected (0.09 sec)

Records: 0 Duplicates: 0 Warnings: 0

mysql> desc Event;

Field	Type	Null	Key	Default	Extra
event_id	int	NO	PRI	NULL	
event_name	text	YES		NULL	
event_date	date	YES		NULL	
event_time	time	YES		NULL	
venue_id	int	YES	MUL	NULL	

Activate Windows
Go to Settings to activate Windows.



Type here to search



22°C



20:15
12-12-2023

Field	Type	Null	Key	Default	Extra
event_id	int	NO	PRI	NULL	
event_name	text	YES		NULL	
event_date	date	YES		NULL	
event_time	time	YES		NULL	
venue_id	int	YES	MUL	NULL	
total_seats	int	YES		NULL	
available_seats	int	YES		NULL	
ticket_price	decimal(10,0)	YES		NULL	
event_type	enum('Movie','Sports','Concert')	YES		NULL	
booking_id	int	YES	MUL	NULL	

10 rows in set (0.00 sec)

mysql> ALTER TABLE Booking

-> ADD CONSTRAINT fk_event

-> FOREIGN KEY (event_id)

-> REFERENCES Event(event_id);

Query OK, 0 rows affected (0.07 sec)

Records: 0 Duplicates: 0 Warnings: 0

mysql> ALTER TABLE Booking

-> ADD CONSTRAINT fk_customer

-> FOREIGN KEY (customer_id)

-> REFERENCES Customer(customer_id);

Query OK, 0 rows affected (0.06 sec)

Records: 0 Duplicates: 0 Warnings: 0

mysql> desc Booking;

Field	Type	Null	Key	Default	Extra
booking_id	int	NO	PRI	NULL	
customer_id	int	YES	MUL	NULL	
event_id	int	YES	MUL	NULL	
num_tickets	int	YES		NULL	
total_cost	decimal(10,0)	YES		NULL	
booking_date	date	YES		NULL	

6 rows in set (0.00 sec)

mysql>

Activate Windows
Go to Settings to activate Windows.



Type here to search



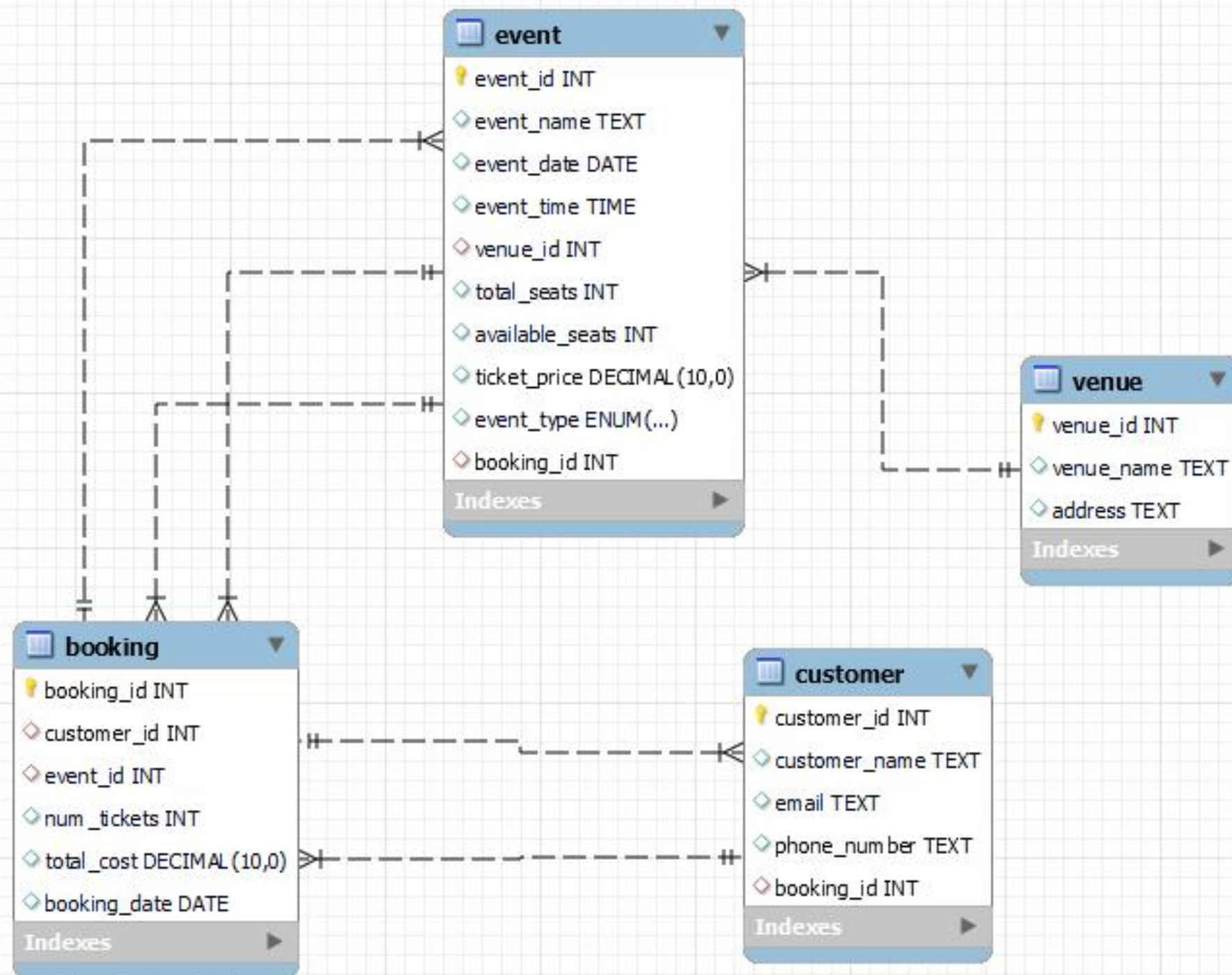
22°C



ENG

20:16
12-12-2023





```
-> (10, 'Amphitheater 10', '707 Open Space, Lucknow, Uttar Pradesh, India'),
-> (11, 'Mall Auditorium 11', '808 Shopper's Delight, Patna, Bihar, India');
ERROR 1146 (42S02): Table 'hmbank.venue' doesn't exist
mysql> use TicketBookingSystem;
Database changed
mysql> /*TASK-2: Select, Where, Between, AND, LIKE:*/
mysql> /* 2.1 Write a SQL query to insert at least 10 sample records into each table*/
mysql> /*Venue*/
mysql> INSERT INTO Venue (venue_id, venue_name, address)
-> VALUES
-> (1, 'Banquet Hall 1', '123 Main Street, Mumbai, Maharashtra, India'),
-> (2, 'Stadium 2', '456 Stadium Road, Delhi, Delhi, India'),
-> (3, 'Convention Center 3', '789 Conference Avenue, Bangalore, Karnataka, India'),
-> (4, 'Auditorium 4', '101 Cultural Street, Kolkata, West Bengal, India'),
-> (5, 'Community Hall 5', '202 Social Lane, Chennai, Tamil Nadu, India'),
-> (6, 'Arena 6', '303 Entertainment Avenue, Hyderabad, Telangana, India'),
-> (7, 'Exhibition Hall 7', '404 Expo Street, Pune, Maharashtra, India'),
-> (8, 'Sports Complex 8', '505 Sports Road, Ahmedabad, Gujarat, India'),
-> (9, 'Cinema Hall 9', '606 Movie Plaza, Jaipur, Rajasthan, India'),
-> (10, 'Amphitheater 10', '707 Open Space, Lucknow, Uttar Pradesh, India'),
-> (11, 'Mall Auditorium 11', '808 Shopper's Delight, Patna, Bihar, India');
Query OK, 11 rows affected (0.03 sec)
Records: 11 Duplicates: 0 Warnings: 0
```

```
mysql> SELECT * FROM VENUE;
```

venue_id	venue_name	address
1	Banquet Hall 1	123 Main Street, Mumbai, Maharashtra, India
2	Stadium 2	456 Stadium Road, Delhi, Delhi, India
3	Convention Center 3	789 Conference Avenue, Bangalore, Karnataka, India
4	Auditorium 4	101 Cultural Street, Kolkata, West Bengal, India
5	Community Hall 5	202 Social Lane, Chennai, Tamil Nadu, India
6	Arena 6	303 Entertainment Avenue, Hyderabad, Telangana, India
7	Exhibition Hall 7	404 Expo Street, Pune, Maharashtra, India
8	Sports Complex 8	505 Sports Road, Ahmedabad, Gujarat, India
9	Cinema Hall 9	606 Movie Plaza, Jaipur, Rajasthan, India
10	Amphitheater 10	707 Open Space, Lucknow, Uttar Pradesh, India
11	Mall Auditorium 11	808 Shopper's Delight, Patna, Bihar, India

```
11 rows in set (0.00 sec)
```

```
mysql> /*EVENT*/
```

```
mysql> INSERT INTO Event (event_name, event_date, event_time, venue_id, total_seats, available_seats, ticket_price, event_type, booking_id)
```

Activate Windows
Go to Settings to activate Windows.



Type here to search



19°C 23:57 12-12-2023


```
mysql>
mysql> SET FOREIGN_KEY_CHECKS=0;
Query OK, 0 rows affected (0.00 sec)

mysql> INSERT INTO Event (event_id, event_name, event_date, event_time, venue_id, total_seats, available_seats, ticket_price, event_type, booking_id)
-> VALUES
-> (101, 'Indian Movie 1', '2023-01-01', '12:00:00', 1, 100, 100, 20.00, 'Movie', 301),
-> (102, 'Indian Sports 1', '2023-02-01', '15:30:00', 2, 150, 150, 30.00, 'Sports', 302),
-> (103, 'Indian Concert 1', '2023-03-01', '18:00:00', 3, 120, 120, 25.00, 'Concert', 303),
-> (104, 'Indian Movie 2', '2023-04-01', '20:00:00', 4, 80, 80, 15.00, 'Movie', 304),
-> (105, 'Indian Sports 2', '2023-05-01', '14:00:00', 5, 200, 200, 40.00, 'Sports', 305),
-> (106, 'Indian Concert 2', '2023-06-01', '17:30:00', 6, 150, 150, 30.00, 'Concert', 306),
-> (107, 'Indian Movie 3', '2023-07-01', '19:00:00', 7, 120, 120, 25.00, 'Movie', 307),
-> (108, 'Indian Sports 3', '2023-08-01', '21:00:00', 8, 100, 100, 20.00, 'Sports', 308),
-> (109, 'Indian Concert 3', '2023-09-01', '16:30:00', 9, 180, 180, 35.00, 'Concert', 309),
-> (110, 'Indian Movie 4', '2023-10-01', '13:00:00', 10, 30, 130, 27.00, 'Movie', 310),
-> (111, 'Indian Sports 4', '2023-11-01', '22:00:00', 11, 90, 90, 18.00, 'Sports', 311);
Query OK, 11 rows affected (0.01 sec)
Records: 11 Duplicates: 0 Warnings: 0
```

```
mysql> select * from event;
```

event_id	event_name	event_date	event_time	venue_id	total_seats	available_seats	ticket_price	event_type	booking_id
101	Indian Movie 1	2023-01-01	12:00:00	1	100	100	20	Movie	301
102	Indian Sports 1	2023-02-01	15:30:00	2	150	150	30	Sports	302
103	Indian Concert 1	2023-03-01	18:00:00	3	120	120	25	Concert	303
104	Indian Movie 2	2023-04-01	20:00:00	4	80	80	15	Movie	304
105	Indian Sports 2	2023-05-01	14:00:00	5	200	200	40	Sports	305
106	Indian Concert 2	2023-06-01	17:30:00	6	150	150	30	Concert	306
107	Indian Movie 3	2023-07-01	19:00:00	7	120	120	25	Movie	307
108	Indian Sports 3	2023-08-01	21:00:00	8	100	100	20	Sports	308
109	Indian Concert 3	2023-09-01	16:30:00	9	180	180	35	Concert	309
110	Indian Movie 4	2023-10-01	13:00:00	10	30	130	27	Movie	310
111	Indian Sports 4	2023-11-01	22:00:00	11	90	90	18	Sports	311

```
11 rows in set (0.00 sec)
```

```
mysql> /*CUSTOMERS*/
```

```
mysql> INSERT INTO Customer (customer_id, customer_name, email, phone_number, booking_id) VALUES
-> (201, 'Rahul Kumar', 'rahul.kumar@example.com', '9876543210', 301),
-> (202, 'Priya Sharma', 'priya.sharma@example.com', '8765432109', 302),
-> (203, 'Amit Singh', 'amit.singh@example.com', '7654321098', 303),
```

Activate Windows
Go to Settings to activate Windows.



Type here to search



24°C

ENG

10:17
13-12-2023

```
mysql> /*Customers*/
mysql> INSERT INTO Customer (customer_id, customer_name, email, phone_number, booking_id) VALUES
```

```
-> (201, 'Rahul Kumar', 'rahul.kumar@example.com', '9876543210', 301),
-> (202, 'Priya Sharma', 'priya.sharma@example.com', '8765432109', 302),
-> (203, 'Amit Singh', 'amit.singh@example.com', '7654321098', 303),
-> (204, 'Neha Patel', 'neha.patel@example.com', '6543210987', 304),
-> (205, 'Raj Malhotra', 'raj.malhotra@example.com', '5432109876', 305),
-> (206, 'Sneha Kapoor', 'sneha.kapoor@example.com', '4321098765', 306),
-> (207, 'Karan Gupta', 'karan.gupta@example.com', '3210987654', 307),
-> (208, 'Pooja Mehta', 'pooja.mehta@example.com', '2109876543', 308),
-> (209, 'Vikas Verma', 'vikas.verma@example.com', '1098765432', 309),
-> (210, 'Anjali Yadav', 'anjali.yadav@example.com', '9876543211', 310),
-> (211, 'Aryan Khanna', 'aryan.khanna@example.com', '8765432101', 311);
```

```
Query OK, 11 rows affected (0.00 sec)
```

```
Records: 11 Duplicates: 0 Warnings: 0
```

```
mysql> select * from customers;
```

```
ERROR 1146 (42S02): Table 'ticketbookingsystem.customers' doesn't exist
```

```
mysql> select * from customer;
```

customer_id	customer_name	email	phone_number	booking_id
201	Rahul Kumar	rahul.kumar@example.com	9876543210	301
202	Priya Sharma	priya.sharma@example.com	8765432109	302
203	Amit Singh	amit.singh@example.com	7654321098	303
204	Neha Patel	neha.patel@example.com	6543210987	304
205	Raj Malhotra	raj.malhotra@example.com	5432109876	305
206	Sneha Kapoor	sneha.kapoor@example.com	4321098765	306
207	Karan Gupta	karan.gupta@example.com	3210987654	307
208	Pooja Mehta	pooja.mehta@example.com	2109876543	308
209	Vikas Verma	vikas.verma@example.com	1098765432	309
210	Anjali Yadav	anjali.yadav@example.com	9876543211	310
211	Aryan Khanna	aryan.khanna@example.com	8765432101	311

```
11 rows in set (0.00 sec)
```

```
mysql> /*Booking*/
```

```
mysql> INSERT INTO Booking (booking_id, customer_id, event_id, num_tickets, total_cost, booking_date) VALUES
```

```
-> (301, 201, 101, 2, 40.00, '2023-01-15'),
-> (302, 202, 102, 3, 90.00, '2023-02-15'),
-> (303, 203, 103, 4, 60.00, '2023-04-15'),
-> (304, 204, 104, 2, 80.00, '2023-05-15'),
-> (305, 205, 105, 3, 90.00, '2023-06-15'),
-> (306, 206, 106, 1, 20.00, '2023-07-15'),
```

Activate Windows
Go to Settings to activate Windows.



Type here to search




```
-> (306, 206, 106, 1, 20.00, '2023-07-15'),
-> (307, 207, 107, 2, 40.00, '2023-08-15'),
-> (308, 208, 108, 1, 25.00, '2023-03-15'),
-> (309, 209, 109, 2, 140.00, '2023-09-15'),
-> (310, 210, 110, 2, 54.00, '2023-10-15'),
-> (311, 211, 111, 3, 54.00, '2023-11-15');
```

Query OK, 11 rows affected (0.01 sec)

Records: 11 Duplicates: 0 Warnings: 0

mysql> select * from booking;

booking_id	customer_id	event_id	num_tickets	total_cost	booking_date
301	201	101	2	40	2023-01-15
302	202	102	3	90	2023-02-15
303	203	103	4	60	2023-04-15
304	204	104	2	80	2023-05-15
305	205	105	3	90	2023-06-15
306	206	106	1	20	2023-07-15
307	207	107	2	40	2023-08-15
308	208	108	1	25	2023-03-15
309	209	109	2	140	2023-09-15
310	210	110	2	54	2023-10-15
311	211	111	3	54	2023-11-15

11 rows in set (0.00 sec)

mysql> /*2.2. Write a SQL query to list all Events.*/

mysql> select * from event;

event_id	event_name	event_date	event_time	venue_id	total_seats	available_seats	ticket_price	event_type	booking_id
101	Indian Movie 1	2023-01-01	12:00:00	1	100	100	20	Movie	301
102	Indian Sports 1	2023-02-01	15:30:00	2	150	150	30	Sports	302
103	Indian Concert 1	2023-03-01	18:00:00	3	120	120	25	Concert	303
104	Indian Movie 2	2023-04-01	20:00:00	4	80	80	15	Movie	304
105	Indian Sports 2	2023-05-01	14:00:00	5	200	200	40	Sports	305
106	Indian Concert 2	2023-06-01	17:30:00	6	150	150	30	Concert	306
107	Indian Movie 3	2023-07-01	19:00:00	7	120	120	25	Movie	307
108	Indian Sports 3	2023-08-01	21:00:00	8	100	100	20	Sports	308
109	Indian Concert 3	2023-09-01	16:30:00	9	180	180	35	Concert	309
110	Indian Movie 4	2023-10-01	13:00:00	10	30	130	27	Movie	310
111	Indian Sports 4	2023-11-01	22:00:00	11	90	90	18	Sports	311

Activate Windows
Go to Settings to activate Windows.



Type here to search




```
mysql> /*2
/*> */
mysql> /*2.3 Write a SQL query to select events with available tickets*/
mysql> SELECT * FROM Event WHERE available_seats > 0;
```

event_id	event_name	event_date	event_time	venue_id	total_seats	available_seats	ticket_price	event_type	booking_id
101	Indian Movie 1	2023-01-01	12:00:00	1	100	100	20	Movie	301
102	Indian Sports 1	2023-02-01	15:30:00	2	150	150	30	Sports	302
103	Indian Concert 1	2023-03-01	18:00:00	3	120	120	25	Concert	303
104	Indian Movie 2	2023-04-01	20:00:00	4	80	80	15	Movie	304
105	Indian Sports 2	2023-05-01	14:00:00	5	200	200	40	Sports	305
106	Indian Concert 2	2023-06-01	17:30:00	6	150	150	30	Concert	306
107	Indian Movie 3	2023-07-01	19:00:00	7	120	120	25	Movie	307
108	Indian Sports 3	2023-08-01	21:00:00	8	100	100	20	Sports	308
109	Indian Concert 3	2023-09-01	16:30:00	9	180	180	35	Concert	309
110	Indian Movie 4	2023-10-01	13:00:00	10	30	130	27	Movie	310
111	Indian Sports 4	2023-11-01	22:00:00	11	90	90	18	Sports	311

11 rows in set (0.00 sec)

```
mysql> /* 2.4 Write a SQL query to select events name partial match with 'cup'*/
mysql> SELECT * FROM Event WHERE event_name LIKE '%cup%';
Empty set (0.00 sec)
```

```
mysql> /* 5. Write a SQL query to select events with ticket price range is between 1000 to 2500*/
mysql> SELECT * FROM Event WHERE ticket_price BETWEEN 1000 AND 2500;
Empty set (0.00 sec)
```

```
mysql> /*6. Write a SQL query to retrieve events with dates falling within a specific range*/
mysql> SELECT * FROM Event
-> WHERE event_date BETWEEN '2023-01-01' AND '2023-12-31';
```

event_id	event_name	event_date	event_time	venue_id	total_seats	available_seats	ticket_price	event_type	booking_id
101	Indian Movie 1	2023-01-01	12:00:00	1	100	100	20	Movie	301
102	Indian Sports 1	2023-02-01	15:30:00	2	150	150	30	Sports	302
103	Indian Concert 1	2023-03-01	18:00:00	3	120	120	25	Concert	303
104	Indian Movie 2	2023-04-01	20:00:00	4	80	80	15	Movie	304
105	Indian Sports 2	2023-05-01	14:00:00	5	200	200	40	Sports	305
106	Indian Concert 2	2023-06-01	17:30:00	6	150	150	30	Concert	306
107	Indian Movie 3	2023-07-01	19:00:00	7	120	120	25	Movie	307
108	Indian Sports 3	2023-08-01	21:00:00	8	100	100	20	Sports	308
109	Indian Concert 3	2023-09-01	16:30:00	9	180	180	35	Concert	309

Activate Windows
Go to Settings to activate Windows.



Type here to search



10:25
13-12-2023

```
mysql> /*7. Write a SQL query to retrieve events with available tickets that also have "Concert" in their
/*> name.
/*> */
```

```
mysql> SELECT * FROM Event
-> WHERE available_seats > 0 AND event_type = 'Concert';
```

event_id	event_name	event_date	event_time	venue_id	total_seats	available_seats	ticket_price	event_type	booking_id
103	Indian Concert 1	2023-03-01	18:00:00	3	120	120	25	Concert	303
106	Indian Concert 2	2023-06-01	17:30:00	6	150	150	30	Concert	306
109	Indian Concert 3	2023-09-01	16:30:00	9	180	180	35	Concert	309

3 rows in set (0.00 sec)

```
mysql> /*8. Write a SQL query to retrieve users in batches of 5, starting from the 6th user*/
```

```
mysql> SELECT * FROM Customer
-> ORDER BY customer_id
-> LIMIT 5 OFFSET 5;
```

customer_id	customer_name	email	phone_number	booking_id
206	Sneha Kapoor	sneha.kapoor@example.com	4321098765	306
207	Karan Gupta	karan.gupta@example.com	3210987654	307
208	Pooja Mehta	pooja.mehta@example.com	2109876543	308
209	Vikas Verma	vikas.verma@example.com	1098765432	309
210	Anjali Yadav	anjali.yadav@example.com	9876543211	310

5 rows in set (0.01 sec)

```
mysql> /*9. Write a SQL query to retrieve bookings details contains booked no of ticket more than 4.*/
```

```
mysql> SELECT * FROM Booking
-> WHERE num_tickets > 4;
Empty set (0.00 sec)
```

```
mysql> /*10.. Write a SQL query to retrieve customer information whose phone number end with '000'
```

```
/*> */
mysql> SELECT * FROM Customer
-> WHERE SUBSTRING(phone_number, -3) = '000';
Empty set (0.00 sec)
```

```
mysql> _
```

Activate Windows
Go to Settings to activate Windows.



Type here to search



24°C



ENG

10:29
13-12-2023



mysql> /*11. Write a SQL query to retrieve the events in order whose seat capacity more than 15000.*/

mysql> SELECT * FROM Event

-> WHERE total_seats > 15000

-> ORDER BY total_seats ASC;

Empty set (0.00 sec)

mysql> /*12. Write a SQL query to select events name not start with 'x', 'y', 'z'*/

mysql> SELECT * FROM Event

-> WHERE NOT (event_name LIKE 'x%' OR event_name LIKE 'y%' OR event_name LIKE 'z%');

event_id	event_name	event_date	event_time	venue_id	total_seats	available_seats	ticket_price	event_type	booking_id
101	Indian Movie 1	2023-01-01	12:00:00	1	100	100	20	Movie	301
102	Indian Sports 1	2023-02-01	15:30:00	2	150	150	30	Sports	302
103	Indian Concert 1	2023-03-01	18:00:00	3	120	120	25	Concert	303
104	Indian Movie 2	2023-04-01	20:00:00	4	80	80	15	Movie	304
105	Indian Sports 2	2023-05-01	14:00:00	5	200	200	40	Sports	305
106	Indian Concert 2	2023-06-01	17:30:00	6	150	150	30	Concert	306
107	Indian Movie 3	2023-07-01	19:00:00	7	120	120	25	Movie	307
108	Indian Sports 3	2023-08-01	21:00:00	8	100	100	20	Sports	308
109	Indian Concert 3	2023-09-01	16:30:00	9	180	180	35	Concert	309
110	Indian Movie 4	2023-10-01	13:00:00	10	30	130	27	Movie	310
111	Indian Sports 4	2023-11-01	22:00:00	11	90	90	18	Sports	311

11 rows in set (0.00 sec)

mysql> /*Tasks 3: Aggregate functions, Having, Order By, GroupBy and Joins:*/

mysql> /*1. Write a SQL query to List Events and Their Average Ticket Prices.*/

mysql> SELECT

-> e.event_id,

-> e.event_name,

-> AVG(e.ticket_price) AS average_ticket_price

-> FROM

-> Event e

-> JOIN

-> Booking b ON e.event_id = b.event_id

-> GROUP BY

-> e.event_id, e.event_name;

event_id	event_name	average_ticket_price
101	Indian Movie 1	20.0000
102	Indian Sports 1	30.0000
103	Indian Concert 1	25.0000

Activate Windows
Go to Settings to activate Windows.



Type here to search



mysql> /*2. Write a SQL query to Calculate the Total Revenue Generated by Events.*/

```
mysql> SELECT
->     e.event_id,
->     e.event_name,
->     SUM(b.total_cost) AS total_revenue
-> FROM
->     Event e
-> JOIN
->     Booking b ON e.event_id = b.event_id
-> GROUP BY
->     e.event_id, e.event_name;
```

event_id	event_name	total_revenue
101	Indian Movie 1	40
102	Indian Sports 1	90
103	Indian Concert 1	60
104	Indian Movie 2	80
105	Indian Sports 2	90
106	Indian Concert 2	20
107	Indian Movie 3	40
108	Indian Sports 3	25
109	Indian Concert 3	140
110	Indian Movie 4	54
111	Indian Sports 4	54

11 rows in set (0.00 sec)

mysql> /*3. Write a SQL query to find the event with the highest ticket sales.*/

```
mysql> SELECT
->     e.event_id,
->     e.event_name,
->     SUM(b.num_tickets) AS total_tickets_sold
-> FROM
->     Event e
-> JOIN
->     Booking b ON e.event_id = b.event_id
-> GROUP BY
->     e.event_id, e.event_name
-> ORDER BY
->     total_tickets_sold DESC
-> LIMIT 1;
```

event_id	event_name	total_tickets_sold
----------	------------	--------------------

Activate Windows
Go to Settings to activate Windows.



Type here to search



25°C



10:36
13-12-2023


```
+-----+-----+-----+
|      103 | Indian Concert 1 |          4 |
+-----+-----+-----+
```

1 row in set (0.00 sec)

mysql> /*4. Write a SQL query to Calculate the Total Number of Tickets Sold for Each Event.

```
mysql> /*> */
mysql> SELECT
->     e.event_id,
->     e.event_name,
->     SUM(b.num_tickets) AS total_tickets_sold
-> FROM
->     Event e
-> JOIN
->     Booking b ON e.event_id = b.event_id
-> GROUP BY
->     e.event_id, e.event_name;
```

```
+-----+-----+-----+
| event_id | event_name      | total_tickets_sold |
+-----+-----+-----+
|      101 | Indian Movie 1  |          2 |
|      102 | Indian Sports 1 |          3 |
|      103 | Indian Concert 1 |          4 |
|      104 | Indian Movie 2  |          2 |
|      105 | Indian Sports 2 |          3 |
|      106 | Indian Concert 2 |          1 |
|      107 | Indian Movie 3  |          2 |
|      108 | Indian Sports 3 |          1 |
|      109 | Indian Concert 3 |          2 |
|      110 | Indian Movie 4  |          2 |
|      111 | Indian Sports 4 |          3 |
+-----+-----+-----+
```

11 rows in set (0.00 sec)

mysql> /*5. Write a SQL query to Find Events with No Ticket Sales*/

```
mysql> SELECT
->     e.event_id,
->     e.event_name
-> FROM
->     Event e
-> LEFT JOIN
->     Booking b ON e.event_id = b.event_id
-> WHERE
->     b.event_id IS NULL;
```

Activate Windows
Go to Settings to activate Windows.



Type here to search



25°C



10:38
13-12-2023



Empty set (0.00 sec)

mysql> /*6. Write a SQL query to Find the User Who Has Booked the Most Tickets*/

```
mysql> SELECT
->     c.customer_id,
->     c.customer_name,
->     COUNT(b.booking_id) AS total_tickets_booked
-> FROM
->     Customer c
-> JOIN
->     Booking b ON c.customer_id = b.customer_id
-> GROUP BY
->     c.customer_id, c.customer_name
-> ORDER BY
->     total_tickets_booked DESC
-> LIMIT 1;
```

customer_id	customer_name	total_tickets_booked
201	Rahul Kumar	1

1 row in set (0.00 sec)

mysql> /*7. Write a SQL query to List Events and the total number of tickets sold for each month.

```
/*> */
mysql> SELECT
->     MONTH(b.booking_date) AS month,
->     e.event_id,
->     e.event_name,
->     SUM(b.num_tickets) AS total_tickets_sold
-> FROM
->     Event e
-> LEFT JOIN
->     Booking b ON e.event_id = b.event_id
-> GROUP BY
->     month, e.event_id, e.event_name
-> ORDER BY
->     month, total_tickets_sold DESC;
```

month	event_id	event_name	total_tickets_sold
1	101	Indian Movie 1	2
2	102	Indian Sports 1	3
3	108	Indian Sports 3	1

Activate Windows
Go to Settings to activate Windows.



Type here to search



25°C

10:39
13-12-2023

month	event_id	event_name	total_tickets_sold
1	101	Indian Movie 1	2
2	102	Indian Sports 1	3
3	108	Indian Sports 3	1
4	103	Indian Concert 1	4
5	104	Indian Movie 2	2
6	105	Indian Sports 2	3
7	106	Indian Concert 2	1
8	107	Indian Movie 3	2
9	109	Indian Concert 3	2
10	110	Indian Movie 4	2
11	111	Indian Sports 4	3

11 rows in set (0.01 sec)

mysql> /*8. Write a SQL query to calculate the average Ticket Price for Events in Each Venue.*/

```
mysql> SELECT
->   v.venue_id,
->   v.venue_name,
->   AVG(e.ticket_price) AS average_ticket_price
-> FROM
->   Venue v
-> JOIN
->   Event e ON v.venue_id = e.venue_id
-> GROUP BY
->   v.venue_id, v.venue_name;
```

venue_id	venue_name	average_ticket_price
1	Banquet Hall 1	20.0000
2	Stadium 2	30.0000
3	Convention Center 3	25.0000
4	Auditorium 4	15.0000
5	Community Hall 5	40.0000
6	Arena 6	30.0000
7	Exhibition Hall 7	25.0000
8	Sports Complex 8	20.0000
9	Cinema Hall 9	35.0000
10	Amphitheater 10	27.0000
11	Mall Auditorium 11	18.0000

11 rows in set (0.10 sec)

Activate Windows
Go to Settings to activate Windows.



Type here to search



11 rows in set (0.10 sec)

mysql> /*9. Write a SQL query to calculate the total Number of Tickets Sold for Each Event Type*/

```
mysql> SELECT
->     e.event_type,
->     SUM(b.num_tickets) AS total_tickets_sold
-> FROM
->     Event e
-> JOIN
->     Booking b ON e.event_id = b.event_id
-> GROUP BY
->     e.event_type
-> ORDER BY
->     total_tickets_sold DESC;
```

event_type	total_tickets_sold
Sports	10
Movie	8
Concert	7

3 rows in set (0.03 sec)

mysql> /*10. Write a SQL query to calculate the total Revenue Generated by Events in Each Year*/

```
mysql> SELECT
->     YEAR(b.booking_date) AS year,
->     SUM(b.total_cost) AS total_revenue
-> FROM
->     Booking b
-> GROUP BY
->     year
-> ORDER BY
->     year;
```

year	total_revenue
2023	693

1 row in set (0.01 sec)

mysql> _

Activate Windows
Go to Settings to activate Windows.



Type here to search



25°C




```
+-----+
1 row in set (0.01 sec)
```

```
mysql> /*11.. Write a SQL query to list users who have booked tickets for multiple events.*/
```

```
mysql> SELECT
->   c.customer_id,
->   c.customer_name,
->   COUNT(DISTINCT b.event_id) AS num_events_booked
-> FROM
->   Customer c
-> JOIN
->   Booking b ON c.customer_id = b.customer_id
-> GROUP BY
->   c.customer_id, c.customer_name
-> HAVING
->   num_events_booked > 1;
Empty set (0.03 sec)
```

```
mysql> /*12. 12. Write a SQL query to calculate the Total Revenue Generated by Events for Each User*/
```

```
mysql> SELECT
->   c.customer_id,
->   c.customer_name,
->   SUM(b.total_cost) AS total_revenue
-> FROM
->   Customer c
-> JOIN
->   Booking b ON c.customer_id = b.customer_id
-> GROUP BY
->   c.customer_id, c.customer_name;
```

customer_id	customer_name	total_revenue
201	Rahul Kumar	40
202	Priya Sharma	90
203	Amit Singh	60
204	Neha Patel	80
205	Raj Malhotra	90
206	Sneha Kapoor	20
207	Karan Gupta	40
208	Pooja Mehta	25
209	Vikas Verma	140
210	Anjali Yadav	54
211	Aryan Khanna	54

Activate Windows
Go to Settings to activate Windows.



Type here to search



25°C



10:43
13-12-2023

11 rows in set (0.00 sec)

mysql> /*13. Write a SQL query to calculate the Average Ticket Price for Events in Each Category and Venue.*/

```
mysql> SELECT
->     e.event_type AS event_category,
->     v.venue_id,
->     v.venue_name,
->     AVG(e.ticket_price) AS average_ticket_price
-> FROM
->     Event e
-> JOIN
->     Venue v ON e.venue_id = v.venue_id
-> GROUP BY
->     e.event_type, v.venue_id, v.venue_name;
```

event_category	venue_id	venue_name	average_ticket_price
Movie	1	Banquet Hall 1	20.0000
Sports	2	Stadium 2	30.0000
Concert	3	Convention Center 3	25.0000
Movie	4	Auditorium 4	15.0000
Sports	5	Community Hall 5	40.0000
Concert	6	Arena 6	30.0000
Movie	7	Exhibition Hall 7	25.0000
Sports	8	Sports Complex 8	20.0000
Concert	9	Cinema Hall 9	35.0000
Movie	10	Amphitheater 10	27.0000
Sports	11	Mall Auditorium 11	18.0000

11 rows in set (0.01 sec)

mysql> /*14.4. Write a SQL query to list Users and the Total Number of Tickets They've Purchased in the Last 30 Days*/

```
mysql> SELECT
->     c.customer_id,
->     c.customer_name,
->     SUM(b.num_tickets) AS total_tickets_purchased
-> FROM
->     Customer c
-> JOIN
->     Booking b ON c.customer_id = b.customer_id
-> WHERE
->     b.booking_date >= CURDATE() - INTERVAL 30 DAY
-> GROUP BY
```

Activate Windows
Go to Settings to activate Windows.



Type here to search



25°C



ENG

10:45
13-12-2023

```
-> WHERE
->   b.booking_date >= CURDATE() - INTERVAL 30 DAY
-> GROUP BY
->   c.customer_id, c.customer_name;
```

```
+-----+-----+-----+
| customer_id | customer_name | total_tickets_purchased |
+-----+-----+-----+
| 211 | Aryan Khanna | 3 |
+-----+-----+-----+
1 row in set (0.01 sec)
```

mysql> /*Tasks 4: Subquery and its types*/

mysql> /*1. Calculate the Average Ticket Price for Events in Each Venue Using a Subquery*/

```
mysql> SELECT
->   v.venue_id,
->   v.venue_name,
->   (SELECT AVG(e.ticket_price) FROM Event e WHERE e.venue_id = v.venue_id) AS average_ticket_price
-> FROM
->   Venue v;
```

```
+-----+-----+-----+
| venue_id | venue_name          | average_ticket_price |
+-----+-----+-----+
| 1 | Banquet Hall 1      | 20.0000              |
| 2 | Stadium 2           | 30.0000              |
| 3 | Convention Center 3 | 25.0000              |
| 4 | Auditorium 4        | 15.0000              |
| 5 | Community Hall 5    | 40.0000              |
| 6 | Arena 6             | 30.0000              |
| 7 | Exhibition Hall 7   | 25.0000              |
| 8 | Sports Complex 8    | 20.0000              |
| 9 | Cinema Hall 9       | 35.0000              |
| 10 | Amphitheater 10     | 27.0000              |
| 11 | Mall Auditorium 11  | 18.0000              |
+-----+-----+-----+
```

11 rows in set (0.00 sec)

mysql> /* 2. Find Events with More Than 50% of Tickets Sold using subquery.*/

```
mysql> SELECT
->   e.event_id,
->   e.event_name
-> FROM
->   Event e
-> WHERE
->   (SELECT COUNT(*) FROM Booking b WHERE b.event_id = e.event_id) > 0.5 * e.total_seats;
```

Activate Windows
Go to Settings to activate Windows.



Type here to search



10:46

13-12-2023

ENG

25°C


```
mysql> /*3. Calculate the Total Number of Tickets Sold for Each Event.*/
```

```
mysql> SELECT
```

```
->     e.event_id,  
->     e.event_name,  
->     (SELECT SUM(b.num_tickets) FROM Booking b WHERE b.event_id = e.event_id) AS total_tickets_sold  
-> FROM  
->     Event e;
```

event_id	event_name	total_tickets_sold
101	Indian Movie 1	2
102	Indian Sports 1	3
103	Indian Concert 1	4
104	Indian Movie 2	2
105	Indian Sports 2	3
106	Indian Concert 2	1
107	Indian Movie 3	2
108	Indian Sports 3	1
109	Indian Concert 3	2
110	Indian Movie 4	2
111	Indian Sports 4	3

```
11 rows in set (0.00 sec)
```

```
mysql> /*4. Find Users Who Have Not Booked Any Tickets Using a NOT EXISTS Subquery*/
```

```
mysql> SELECT
```

```
->     c.customer_id,  
->     c.customer_name  
-> FROM  
->     Customer c  
-> WHERE  
->     NOT EXISTS (SELECT 1 FROM Booking b WHERE b.customer_id = c.customer_id);
```

```
Empty set (0.00 sec)
```

```
mysql> /*5. List Events with No Ticket Sales Using a NOT IN Subquery.
```

```
/*> */
```

```
mysql> SELECT
```

```
->     event_id,  
->     event_name  
-> FROM  
->     Event  
-> WHERE  
->     event_id NOT IN (SELECT DISTINCT event_id FROM Booking);
```

Activate Windows
Go to Settings to activate Windows.



Type here to search



25°C



10:48
13-12-2023

```
mysql> /*Calculate the Total Number of Tickets Sold for Each Event Type Using a Subquery in the FROM
```

```
/*> Clause*/
```

```
mysql> SELECT
->     et.event_type,
->     SUM(b.num_tickets) AS total_tickets_sold
-> FROM
->     (SELECT DISTINCT event_type FROM Event) et
-> LEFT JOIN
->     Booking b ON b.event_id IN (SELECT event_id FROM Event WHERE event_type = et.event_type)
-> GROUP BY
->     et.event_type;
```

event_type	total_tickets_sold
Movie	8
Sports	10
Concert	7

```
3 rows in set (0.01 sec)
```

```
mysql> /*7. Find Events with Ticket Prices Higher Than the Average Ticket Price Using a Subquery in the
```

```
/*> WHERE Clause.*/
```

```
mysql> SELECT
->     event_id,
->     event_name,
->     ticket_price
-> FROM
->     Event
-> WHERE
->     ticket_price > (SELECT AVG(ticket_price) FROM Event);
```

event_id	event_name	ticket_price
102	Indian Sports 1	30
105	Indian Sports 2	40
106	Indian Concert 2	30
109	Indian Concert 3	35
110	Indian Movie 4	27

```
5 rows in set (0.00 sec)
```

```
mysql>
```

Activate Windows
Go to Settings to activate Windows.



Type here to search



25°C



10:49
13-12-2023



5 rows in set (0.00 sec)

mysql> /*8. Calculate the Total Revenue Generated by Events for Each User Using a Correlated Subquery*/

mysql> SELECT

```
-> c.customer_id,  
-> c.customer_name,  
-> (SELECT SUM(b.total_cost) FROM Booking b WHERE b.customer_id = c.customer_id) AS total_revenue  
-> FROM  
-> Customer c;
```

customer_id	customer_name	total_revenue
201	Rahul Kumar	40
202	Priya Sharma	90
203	Amit Singh	60
204	Neha Patel	80
205	Raj Malhotra	90
206	Sneha Kapoor	20
207	Karan Gupta	40
208	Pooja Mehta	25
209	Vikas Verma	140
210	Anjali Yadav	54
211	Aryan Khanna	54

11 rows in set (0.00 sec)

mysql> /*9. List Users Who Have Booked Tickets for Events in a Given Venue Using a Subquery in the WHERE

/*> Clause.*/

mysql> SELECT

```
-> c.customer_id,  
-> c.customer_name  
-> FROM  
-> Customer c  
-> WHERE  
-> EXISTS (  
-> SELECT 1  
-> FROM Booking b  
-> JOIN Event e ON b.event_id = e.event_id  
-> WHERE b.customer_id = c.customer_id AND e.venue_id =1);
```

customer_id	customer_name
201	Rahul Kumar

Activate Windows
Go to Settings to activate Windows.



Type here to search



25°C

10:51
13-12-2023

11 rows in set (0.01 sec)

mysql> /*10. Calculate the Total Number of Tickets Sold for Each Event Category Using a Subquery with

/*> GROUP BY*/

mysql> SELECT

```
-> e.event_type AS event_category,  
-> SUM(b.num_tickets) AS total_tickets_sold  
-> FROM  
-> Event e  
-> JOIN  
-> Booking b ON e.event_id = b.event_id  
-> GROUP BY  
-> e.event_type;
```

event_category	total_tickets_sold
Movie	8
Sports	10
Concert	7

3 rows in set (0.00 sec)

mysql> /*11. Find Users Who Have Booked Tickets for Events in each Month Using a Subquery with

/*> DATE_FORMAT.

/*> */

mysql> SELECT

```
-> c.customer_id,  
-> c.customer_name,  
-> DATE_FORMAT(b.booking_date, '%Y-%m') AS booking_month  
-> FROM  
-> Customer c  
-> JOIN  
-> Booking b ON c.customer_id = b.customer_id  
-> GROUP BY  
-> c.customer_id, c.customer_name, booking_month;
```

customer_id	customer_name	booking_month
201	Rahul Kumar	2023-01
202	Priya Sharma	2023-02
203	Amit Singh	2023-04
204	Neha Patel	2023-05
205	Raj Malhotra	2023-06
206	Sneha Kapoor	2023-07

Activate Windows
Go to Settings to activate Windows.



Type here to search



25°C

10:53
13-12-2023

customer_id	customer_name	booking_month
201	Rahul Kumar	2023-01
202	Priya Sharma	2023-02
203	Amit Singh	2023-04
204	Neha Patel	2023-05
205	Raj Malhotra	2023-06
206	Sneha Kapoor	2023-07
207	Karan Gupta	2023-08
208	Pooja Mehta	2023-03
209	Vikas Verma	2023-09
210	Anjali Yadav	2023-10
211	Aryan Khanna	2023-11

11 rows in set (0.00 sec)

mysql> /*12. Calculate the Average Ticket Price for Events in Each Venue Using a Subquery*/

```
mysql> SELECT
->     v.venue_id,
->     v.venue_name,
->     (SELECT AVG(e.ticket_price) FROM Event e WHERE e.venue_id = v.venue_id) AS average_ticket_price
-> FROM
->     Venue v;
```

venue_id	venue_name	average_ticket_price
1	Banquet Hall 1	20.0000
2	Stadium 2	30.0000
3	Convention Center 3	25.0000
4	Auditorium 4	15.0000
5	Community Hall 5	40.0000
6	Arena 6	30.0000
7	Exhibition Hall 7	25.0000
8	Sports Complex 8	20.0000
9	Cinema Hall 9	35.0000
10	Amphitheater 10	27.0000
11	Mall Auditorium 11	18.0000

11 rows in set (0.00 sec)

mysql> _

Activate Windows
Go to Settings to activate Windows.



Type here to search



10:53
13-12-2023