

MySQL Workbench

Local instance MySQL80

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

- afs
- apple
- bheem
- city
- cricket
- dtb
- dtcc
- eventticketingsystem
 - Tables
 - Views
 - Stored Procedures
 - Functions
- fred
- frnk
- games
- hotel
- hotell
- india
- india_db
- india_dbc

Administration Schemas

Information

No object selected

Object Info Session

File 35* SQL File 36* SQL File 37* SQL File 38* SQL File 39* SQL File 40* SQL File 41* SQL File 42* SQL File 43* SQL File 44* SQL File 45* SQL File 46*

Limit to 1000 rows

```
1
2  -- Create Database
3
4  CREATE DATABASE EventTicketingSystem;
5  USE EventTicketingSystem;
6
7  -- Create Tables
8
9
10 -- Events Table
11 CREATE TABLE Events (
12     EventID INT AUTO_INCREMENT PRIMARY KEY,
13     EventName VARCHAR(100),
14     EventDate DATE,
15     Location VARCHAR(100),
16     TotalTickets INT
17 );
18
19 -- Customers Table
```

Output

Action Output

#	Time	Action	Message	Duration / Fetch
1	16:05:40	CREATE DATABASE EventTicketingSystem	1 row(s) affected	0.000 sec
2	16:05:40	USE EventTicketingSystem	0 row(s) affected	0.000 sec

MySQL Workbench

Local instance MySQL80

File Edit View Query Database Server Tools Scripting Help

Navigator

Filter objects

SCHEMAS

- afs
- apple
- bheem
- city
- cricket
- dtb
- dtcc
- eventticketingsystem**
 - Tables
 - Views
 - Stored Procedures
 - Functions
- frnd
- frnk
- games
- hotel
- hotell
- india
- india_db
- india_db

Administration Schemas

Information

No object selected

Object Info Session

File 35* SQL File 36* SQL File 37* SQL File 38* SQL File 39* SQL File 40* SQL File 41* SQL File 42* SQL File 43* SQL File 44* SQL File 45* SQL File 46*

Limit to 1000 rows

```
6
7 -- Creating Tables
8 -- Events Table
9 CREATE TABLE Events (
10     EventID INT AUTO_INCREMENT PRIMARY KEY,
11     EventName VARCHAR(100),
12     EventDate DATE,
13     Location VARCHAR(100),
14     TotalTickets INT
15 );
16 desc events;
17 -- Customers Table
```

Result Grid

Field	Type	Nul	Key	Default	Extra
EventID	int	NO	PRI	NULL	auto_increment
EventName	varchar(100)	YES		NULL	
EventDate	date	YES		NULL	
Location	varchar(100)	YES		NULL	
TotalTickets	int	YES		NULL	

Result 1

Output

Action Output

#	Time	Action	Message	Duration / Fetch
2	16:05:40	USE EventTicketingSystem	0 row(s) affected	0.000 sec
3	16:09:28	CREATE TABLE Events (EventID INT AUTO_INCREMENT PRIMARY KEY, Event...	0 row(s) affected	0.031 sec
4	16:09:28	desc events	5 row(s) returned	0.000 sec / 0.000 sec

Local instances MySQL80 x

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

- o's
- apple
- bheem
- city
- cricket
- dtb
- dtcc
- eventticketingsystem
 - Tables
 - customers
 - Columns
 - Indexes
 - Foreign Keys
 - Triggers
 - events
 - Views
 - Stored Procedures
 - Functions
- frnd
- frnk

Administration Schemas

Information

Schema: eventticketingsystem

File 35* SQL File 36* SQL File 37* SQL File 38* SQL File 39* SQL File 40* SQL File 41* SQL File 42* SQL File 43* SQL File 44* SQL File 45* SQL File 46*

Limit to 1000 rows

```
13 Location VARCHAR(100),
14 TotalTickets INT
15 };
16 desc events;
17 -- Customers Table
18 CREATE TABLE Customers (
19 CustomerID INT AUTO_INCREMENT PRIMARY KEY,
20 CustomerName VARCHAR(100),
21 Email VARCHAR(100)
22 );
23 desc customers;
24 -- Tickets Table
```

Result Grid

Field	Type	Null	Key	Default	Extra
CustomerID	int	NO	PR	<u>NULL</u>	auto_increment
CustomerName	varchar(100)	YES		<u>NULL</u>	
Email	varchar(100)	YES		<u>NULL</u>	

Result 2 x

Output

Action Output

#	Time	Action	Message	Duration / Fetch
4	16:09:28	desc events	5 row(s) returned	0.000 sec / 0.000 sec
5	16:13:16	CREATE TABLE Customers (CustomerID INT AUTO_INCREMENT PRIMARY KEY,	0 row(s) affected	0.015 sec
6	16:13:16	desc customers	3 row(s) returned	0.000 sec / 0.000 sec

Object Info Session

MySQL Workbench

Local instance MySQL80

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

- afs
- apple
- bheem
- city
- cricket
- dtb
- dtcc
- eventticketingsystem
 - Tables
 - customers
 - Columns
 - Indexes
 - Foreign Keys
 - Triggers
 - events
 - tickets
 - Columns
 - Indexes
 - Foreign Keys
 - Triggers

Administration Schemas

Information

Schema: eventticketingsystem

Object Info Session

File 35* SQL File 36* SQL File 37* SQL File 38* SQL File 39* SQL File 40* SQL File 41* SQL File 42* SQL File 43* SQL File 44* SQL File 45* SQL File 46*

Limit to 1000 rows

```
23 desc customers;
24 -- Tickets Table
25 CREATE TABLE Tickets (
26     TicketID INT AUTO_INCREMENT PRIMARY KEY,
27     EventID INT,
28     TicketType VARCHAR(50),
29     Price DECIMAL(8,2),
30     Status VARCHAR(20) DEFAULT 'Available', -- Available, Sold, Reserved
31     FOREIGN KEY (EventID) REFERENCES Events(EventID)
32 );
33 desc tickets;
34
```

Result Grid

Field	Type	Null	Key	Default	Extra
TicketID	int	NO	PRI	<small>NULL</small>	auto_increment
EventID	int	YES	MUL	<small>NULL</small>	
TicketType	varchar(50)	YES		<small>NULL</small>	
Price	decimal(8,2)	YES		<small>NULL</small>	
Status	varchar(20)	YES		Available	

Result 3 >>

Output

Action Output

#	Time	Action	Message	Duration / Fetch
✓ 6	15:13:16	desc customers	3 row(s) returned	0.000 sec / 0.000 sec
✓ 7	15:15:49	CREATE TABLE Tickets (TicketID INT AUTO_INCREMENT PRIMARY KEY, Even...	0 row(s) affected	0.032 sec
✓ 8	15:15:49	desc tickets	5 row(s) returned	0.015 sec / 0.000 sec

MySQL Workbench

Local instance MySQL80

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects:

- afs
- apple
- bheem
- city
- cricket
- ddb
- dtcc
- eventticketingsystem
 - Tables
 - customers
 - Columns
 - Indexes
 - Foreign Keys
 - events
 - reservations
 - Columns
 - Indexes
 - Foreign Keys
 - Triggers

Administration Schemas

Information

Schema: eventticketingsystem

Object Info Session

SQL File 35* SQL File 36* SQL File 37* SQL File 38* SQL File 39* SQL File 40* SQL File 41* SQL File 42* SQL File 43* SQL File 44* SQL File 45* SQL File 46*

Limit to 1000 rows

```
35 -- Reservations Table
36 CREATE TABLE Reservations (
37     ReservationID INT AUTO_INCREMENT PRIMARY KEY,
38     CustomerID INT,
39     TicketID INT,
40     ReservationDate DATE,
41     FOREIGN KEY (CustomerID) REFERENCES Customers(CustomerID),
42     FOREIGN KEY (TicketID) REFERENCES Tickets(TicketID)
43 );
44 desc Reservations;
45
46 -- Sales Table
```

Result Grid

Field	Type	Null	Key	Default	Extra
ReservationID	int	NO	PRI	NULL	auto_increment
CustomerID	int	YES	MUL	NULL	
TicketID	int	YES	MUL	NULL	
ReservationDate	date	YES		NULL	

Result 4 x

Output

Action Output

#	Time	Action	Message	Duration / Fetch
8	16:15:49	desc tickets	5 row(s) returned	0.015 sec / 0.000 sec
9	16:16:46	CREATE TABLE Reservations (ReservationID INT AUTO_INCREMENT PRIMARY K...	0 row(s) affected	0.031 sec
10	16:16:46	desc Reservations	4 row(s) returned	0.000 sec / 0.000 sec

MySQL Workbench

Local instance MySQL80

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

dtb

dtcc

eventticketingsystem

Tables

customers

Columns

Indexes

Foreign Keys

Triggers

events

reservations

Columns

Indexes

Foreign Keys

Triggers

sales

Columns

Indexes

Foreign Keys

Triggers

Administration Schemas

Information

Schema: eventticketingsystem

Object Info Session

File 35* SQL File 36* SQL File 37* SQL File 38* SQL File 39* SQL File 40* SQL File 41* SQL File 42* SQL File 43* SQL File 44* SQL File 45* SQL File 46*

Limit to 1000 rows

```
-- Sales Table
CREATE TABLE Sales (
  SaleID INT AUTO_INCREMENT PRIMARY KEY,
  CustomerID INT,
  TicketID INT,
  SaleDate DATE,
  Amount DECIMAL(8,2),
  FOREIGN KEY (CustomerID) REFERENCES Customers(CustomerID),
  FOREIGN KEY (TicketID) REFERENCES Tickets(TicketID)
);
desc sales;
```

Result Grid

Field	Type	Nul	Key	Default	Extra
SaleID	int	NO	PRI	<u>NULL</u>	auto_increment
CustomerID	int	YES	MUL	<u>NULL</u>	
TicketID	int	YES	MUL	<u>NULL</u>	
SaleDate	date	YES		<u>NULL</u>	
Amount	decimal(8,2)	YES		<u>NULL</u>	

Result 5

Output

Action Output

#	Time	Action	Message	Duration / Fetch
✓ 10	16:16:46	desc Reservations	4 row(s) returned	0.000 sec / 0.000 sec
✓ 11	16:17:25	CREATE TABLE Sales (SaleID INT AUTO_INCREMENT PRIMARY KEY, Customer...	0 row(s) affected	0.016 sec
✓ 12	16:17:25	desc sales	5 row(s) returned	0.000 sec / 0.000 sec

Local instance MySQL80 x

File Edit View Query Database Server Tools Scripting Help

Navigator: Local instance MySQL80 x

SCHEMAS

Filter objects

dtb
dtcc
eventticketingsystem

Tables

- customers
 - Columns
 - Indexes
 - ForeignKeys
 - Triggers
- events
 - Columns
 - Indexes
 - ForeignKeys
 - Triggers
- reservations
 - Columns
 - Indexes
 - ForeignKeys
 - Triggers
- sales
 - Columns
 - Indexes
 - ForeignKeys
 - Triggers

Administration Schemas

Information

Schema: eventticketingsystem

Object Info Session

File 35* SQL File 36* SQL File 37* SQL File 38* SQL File 39* SQL File 40* SQL File 41* SQL File 42* SQL File 43* SQL File 44* SQL File 45* SQL File 46*

Limit to 1000 rows

```
61
62 -- Insert into Events
63 INSERT INTO Events (EventName, EventDate, Location, TotalTickets) VALUES
64 ('Music Concert', '2025-05-10', 'vellachery', 100),
65 ('Cricket Match', '2025-06-15', 'Gundy', 50),
66 ('Food Festival', '2025-05-22', 'Adyar', 200);
67 select * from events;
68
69 -- Insert into Customers
70 INSERT INTO Customers (CustomerName, Email) VALUES
71 ('Korkai', 'korkaimaran@gmail.com'),
72 ('Surya Rao', 'suryarao@gmail.com');
```

Result Grid

EventID	EventName	EventDate	Location	TotalTickets
1	Music Concert	2025-05-10	vellachery	100
2	Cricket Match	2025-06-15	Gundy	50
3	Food Festival	2025-05-22	Adyar	200
* null	* null	* null	* null	* null

events 6 x

Output

Action Output

#	Time	Action	Message	Duration / Fetch
✓ 12	16:17:25	desc sales	5 row(s) returned	0.000 sec / 0.000 sec
✓ 13	16:18:39	INSERT INTO Events (EventName, EventDate, Location, TotalTickets) VALUES ('Music ...	3 row(s) affected Records: 3 Duplicates: 0 Warnings: 0	0.000 sec
✓ 14	16:18:39	select * from events LIMIT 0, 1000	3 row(s) returned	0.000 sec / 0.000 sec

Local instance MySQL80 x

File Edit View Query Database Server Tools Scripting Help

Navigator: .File 35* SQL File 36* SQL File 37* SQL File 38* SQL File 39* SQL File 40* SQL File 41* SQL File 42* SQL File 43* SQL File 44* SQL File 45* SQL File 46*

SCHEMAS

Filter objects

- dtb
- dtcc
- eventticketingsystem**
 - Tables
 - customers
 - Columns
 - Indexes
 - ForeignKeys
 - Triggers
 - events
 - reservations
 - Columns
 - Indexes
 - ForeignKeys
 - Triggers
 - sales
 - Columns
 - Indexes
 - ForeignKeys
 - Triggers
- Administration
- Schemas

Information

Schema: eventticketingsystem

Object Info Session

```
67 • select * from events;
68
69 -- Insert into Customers
70 • INSERT INTO Customers (CustomerName, Email) VALUES
71 ('Korkai', 'korkaimaran@gmail.com'),
72 ('Surya Rao', 'suryaraom@gmail.com'),
73 ('Hari', 'harikrishnan@gmail.com');
74 • select * from customers;
75
76 -- Insert into Tickets
77 • INSERT INTO Tickets (EventID, TicketType, Price) VALUES
78 (1, 'VIP', 150.00);
```

Result Grid

CustomerID	CustomerName	Email
1	Korkai	korkaimaran@gmail.com
2	Surya Rao	suryaraom@gmail.com
3	Hari	harikrishnan@gmail.com
NULL	NULL	NULL

customers 7 x

Output

Action Output

#	Time	Action	Message	Duration / Fetch
✓ 14	16:18:39	select * from events LIMIT 0, 1000	3 row(s) returned	0.000 sec / 0.000 sec
✓ 15	16:19:16	INSERT INTO Customers (CustomerName, Email) VALUES ('Korkai', 'korkaimaran@gmail.com'), ('Surya Rao', 'suryaraom@gmail.com'), ('Hari', 'harikrishnan@gmail.com');	3 row(s) affected Records: 3 Duplicates: 0 Warnings: 0	0.016 sec
✓ 16	16:19:16	select * from customers LIMIT 0, 1000	3 row(s) returned	0.000 sec / 0.000 sec

Local instance MySQL80

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

dtb
dtcc
eventticketingsystem

Tables

customers
Columns
Indexes
Foreign Keys
Triggers
events
reservations
Columns
Indexes
Foreign Keys
Triggers
sales
Columns
Indexes
Foreign Keys
Triggers

Administration Schemas

Information

Schema: eventticketingsystem

Object Info Session

File 35* SQL File 36* SQL File 37* SQL File 38* SQL File 39* SQL File 40* SQL File 41* SQL File 42* SQL File 43* SQL File 44* SQL File 45* SQL File 46*

Limit to 1000 rows

```
73 ('Hari', 'harikrishnan@gmail.com');
74 select * from customers;
75
76 -- Insert into Tickets
77 INSERT INTO Tickets (EventID, TicketType, Price) VALUES
78 (1, 'VIP', 150.00),
79 (1, 'Regular', 80.00),
80 (2, 'Regular', 50.00),
81 (3, 'VIP', 200.00),
82 (3, 'Regular', 100.00);
83 select * from tickets;
84
```

Result Grid

TicketID	EventID	TicketType	Price	Status
1	1	VIP	150.00	Available
2	1	Regular	80.00	Available
3	2	Regular	50.00	Available
4	3	VIP	200.00	Available
5	3	Regular	100.00	Available

tickets 0

Output

Action Output

#	Time	Action	Message	Duration / Fetch
✓ 16	15:19:16	select * from customers LIMIT 0, 1000	3 row(s) returned	0.000 sec / 0.000 sec
✓ 17	15:19:59	INSERT INTO Tickets (EventID, TicketType, Price) VALUES (1, 'VIP', 150.00), (1, 'Regul...	5 row(s) affected Records: 5 Duplicates: 0 Warnings: 0	0.016 sec
✓ 18	15:19:59	select * from tickets LIMIT 0, 1000	5 row(s) returned	0.000 sec / 0.000 sec

Local instance MySQL80

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

dtb

dtcc

eventticketingsystem

Tables

customers

Columns

Indexes

Foreign Keys

Triggers

events

reservations

Columns

Indexes

Foreign Keys

Triggers

sales

Columns

Indexes

Foreign Keys

Triggers

Administration Schemas

File 35* SQL File 36* SQL File 37* SQL File 38* SQL File 39* SQL File 40* SQL File 41* SQL File 42* SQL File 43* SQL File 44* SQL File 45* SQL File 46*

Limit to 1000 rows

82 (3, 'Regular', 100.00);

83 select * from tickets;

84

85 -- Insert into Reservations

86 INSERT INTO Reservations (CustomerID, TicketID, ReservationDate) VALUES

87 (1, 1, '2025-04-01'),

88 (2, 3, '2025-04-03');

89 select * from reservations;

90

91 -- Insert into Sales

92 INSERT INTO Sales (CustomerID, TicketID, SaleDate, Amount) VALUES

93 (1, 2, '2025-04-05', 80.00);

Result Grid

Filter Rows

Edits

Export/Import

Wrap Cell Content

ReservationID	CustomerID	TicketID	ReservationDate
1	1	1	2025-04-01
2	2	3	2025-04-03
NULL	NULL	NULL	NULL

reservations 9

Apply

Output

Action Output

#	Time	Action	Message	Duration / Fetch
✓ 18	16:19:59	select * from tickets LIMIT 0, 1000	5 row(s) returned	0.000 sec / 0.000 sec
✓ 19	16:22:00	INSERT INTO Reservations (CustomerID, TicketID, ReservationDate) VALUES (1, 1, '20...	2 row(s) affected Records: 2 Duplicates: 0 Warnings: 0	0.000 sec
✓ 20	16:22:17	select * from reservations LIMIT 0, 1000	2 row(s) returned	0.000 sec / 0.000 sec

Object Info Session

Local instance MySQL80 x

File Edit View Query Database Server Tools Scripting Help

Navigator: Local instance MySQL80 x

SCHEMAS

Filter objects

dtb
dtcc
eventticketingsystem

Tables

- customers
 - Columns
 - Indexes
 - Foreign Keys
 - Triggers
- events
 - Columns
 - Indexes
 - Foreign Keys
 - Triggers
- reservations
 - Columns
 - Indexes
 - Foreign Keys
 - Triggers
- sales
 - Columns
 - Indexes
 - Foreign Keys
 - Triggers

Administration Schemas

Information

Schema: eventticketingsystem

Object Info Session

File 35* SQL File 36* SQL File 37* SQL File 38* SQL File 39* SQL File 40* SQL File 41* SQL File 42* SQL File 43* SQL File 44* SQL File 45* SQL File 46*

Limit to 1000 rows

```
88 (2, 3, '2025-04-03');
89 select * from reservations;
90
91 -- Insert into Sales
92 INSERT INTO Sales (CustomerID, TicketID, SaleDate, Amount) VALUES
93 (1, 2, '2025-04-05', 80.00),
94 (2, 4, '2025-04-10', 200.00),
95 (3, 5, '2025-04-12', 100.00);
96 select * from sales;
97
98 -- Update sold tickets' status
99 UPDATE Tickets
```

Result Grid

	SaleID	CustomerID	TicketID	SaleDate	Amount
1	1	1	2	2025-04-05	80.00
2	2	2	4	2025-04-10	200.00
3	3	3	5	2025-04-12	100.00
4	NULL	NULL	NULL	NULL	NULL

sales 10 x

Output

Action Output

#	Time	Action	Message	Duration / Fetch
20	16:22:17	select * from reservations LIMIT 0, 1000	2 row(s) returned	0.000 sec / 0.000 sec
21	16:22:57	INSERT INTO Sales (CustomerID, TicketID, SaleDate, Amount) VALUES (1, 2, '2025-04-05', 80.00), (2, 4, '2025-04-10', 200.00), (3, 5, '2025-04-12', 100.00);	3 row(s) affected Records: 3 Duplicates: 0 Warnings: 0	0.015 sec
22	16:22:58	select * from sales LIMIT 0, 1000	3 row(s) returned	0.000 sec / 0.000 sec

Local instance MySQL80 x

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

dbt
dtcc
eventticketingsystem

Tables

customers
Columns
Indexes
Foreign Keys
Triggers
events
reservations
Columns
Indexes
Foreign Keys
Triggers
sales
Columns
Indexes
Foreign Keys
Triggers

Administration Schemas

Information

Schema:
eventticketingsystem

Object Info Session

File 35* SQL File 36* SQL File 37* SQL File 38* SQL File 39* SQL File 40* SQL File 41* SQL File 42* SQL File 43* SQL File 44* SQL File 45* SQL File 46*

Limit to 1000 rows

```
94 (2, 4, '2025-04-10', 200.00),
95 (3, 5, '2025-04-12', 100.00);
96 select * from sales;
97
98 -- Update sold tickets' status
99 UPDATE Tickets
100 SET Status = 'Sold'
101 WHERE TicketID IN (2, 4, 5);
102 select * from tickets;
103
104 -- Update reserved tickets' status
105 UPDATE Tickets
```

Result Grid

TicketID	EventID	TicketType	Price	Status
1	1	VIP	150.00	Available
2	1	Regular	80.00	Sold
3	2	Regular	50.00	Available
4	3	VIP	200.00	Sold
5	3	Regular	100.00	Sold

tickets i1 x

Output

Action Output

#	Time	Action	Message	Duration / Fetch
22	16:22:58	select * from sales LIMIT 0. 1000	3 row(s) returned	0.000 sec / 0.000 sec
23	16:26:17	UPDATE Tickets SET Status = 'Sold' WHERE TicketID IN (2, 4, 5)	3 row(s) affected Rows matched: 3 Changed: 3 Warnings: 0	0.015 sec
24	16:26:17	select * from tickets LIMIT 0. 1000	5 row(s) returned	0.000 sec / 0.000 sec

MySQL Workbench

Local instance MySQL80

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

dtb

dtcc

eventticketingsystem

Tables

customers

Columns

Indexes

Foreign Keys

Triggers

events

reservations

Columns

Indexes

Foreign Keys

Triggers

sales

Columns

Indexes

Foreign Keys

Triggers

Administration Schemas

Information

Schema: eventticketingsystem

Object Info Session

File 35* SQL File 36* SQL File 37* SQL File 38* SQL File 39* SQL File 40* SQL File 41* SQL File 42* SQL File 43* SQL File 44* SQL File 45*

Limit to 1000 rows

```
100 SET Status = 'Sold'
101 WHERE TicketID IN (2, 4, 5);
102 select * from tickets;
103
104 -- Update reserved tickets' status
105 UPDATE Tickets
106 SET Status = 'Reserved'
107 WHERE TicketID IN (1, 3);
108 select * from tickets;
109
110 -----
111 -- Queries
```

Result Grid

TicketID	EventID	TicketType	Price	Status
1	1	VIP	150.00	Reserved
2	1	Regular	80.00	Sold
3	2	Regular	50.00	Reserved
4	3	VIP	200.00	Sold
5	3	Regular	100.00	Sold

tickets 12 x

Output

Action Output

#	Time	Action	Message	Duration / Fetch
✓ 24	16:26:17	select * from tickets LIMIT 0, 1000	5 row(s) returned	0.000 sec / 0.000 sec
✓ 25	16:27:25	UPDATE Tickets SET Status = 'Reserved' WHERE TicketID IN (1, 3)	2 row(s) affected Rows matched: 2 Changed: 2 Warnings: 0	0.016 sec
✓ 26	16:27:25	select * from tickets LIMIT 0, 1000	5 row(s) returned	0.000 sec / 0.000 sec

Local instance MySQL80

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

db
dtcc
eventticketingsystem

Tables

- customers
 - Columns
 - Indexes
 - Foreign Keys
 - Triggers
- events
- reservations
 - Columns
 - Indexes
 - Foreign Keys
 - Triggers
- sales
 - Columns
 - Indexes
 - Foreign Keys
 - Triggers

Administration Schemas

Information

Schema: eventticketingsystem

Object Info Session

SQL File 35* SQL File 36* SQL File 37* SQL File 38* SQL File 39* SQL File 40* SQL File 41* SQL File 42* SQL File 43* SQL File 44* SQL File 45* SQL File 46*

Limit to 1000 rows

```
-- 1. JOIN: List customer names and the events they reserved or bought tickets for
SELECT Customers.CustomerName, Events.EventName, Tickets.TicketType, Tickets.Price
FROM Customers
JOIN Reservations ON Customers.CustomerID = Reservations.CustomerID
JOIN Tickets ON Reservations.TicketID = Tickets.TicketID
JOIN Events ON Tickets.EventID = Events.EventID
UNION ALL
SELECT Customers.CustomerName, Events.EventName, Tickets.TicketType, Tickets.Price
FROM Customers
JOIN Sales ON Customers.CustomerID = Sales.CustomerID
JOIN Tickets ON Sales.TicketID = Tickets.TicketID
JOIN Events ON Tickets.EventID = Events.EventID
```

Result Grid

	CustomerName	EventName	TicketType	Price
▶	Korkai	Music Concert	VIP	150.00
	Surya Rao	Cricket Match	Regular	50.00
	Korkai	Music Concert	Regular	80.00
	Surya Rao	Food Festival	VIP	200.00
	Hari	Food Festival	Regular	100.00

Result 13 x Read Only

Output

Action Output

#	Time	Action	Message	Duration / Fetch
✓ 25	16:27:25	UPDATE Tickets SET Status = 'Reserved' WHERE TicketID IN (1, 3)	2 row(s) affected Rows matched: 2 Changed: 2 Warnings: 0	0.016 sec
✓ 26	16:27:25	select * from tickets LIMIT 0, 1000	5 row(s) returned	0.000 sec / 0.000 sec
✓ 27	16:34:08	SELECT Customers.CustomerName, Events.EventName, Tickets.TicketType, Tickets.Pr...	5 row(s) returned	0.000 sec / 0.000 sec

mysql workbench

Local instance MySQL80 x

File Edit View Query Database Server Tools Scripting Help

Navigation: File 35* SQL File 36* SQL File 37* SQL File 38* SQL File 39* SQL File 40* SQL File 41* SQL File 42* SQL File 43* SQL File 44* SQL File 45* SQL File 46*

SCHEMAS

Filter objects

- dtb
- dtcc
- eventticketingsystem
 - Tables
 - customers
 - Columns
 - Indexes
 - ForeignKeys
 - Triggers
 - events
 - reservations
 - Columns
 - Indexes
 - ForeignKeys
 - Triggers
 - sales
 - Columns
 - Indexes
 - ForeignKeys
 - Triggers
- Administration
- Schemas

Schema: eventticketingsystem

```
-- 2. SUBQUERY: Find customers who bought VIP tickets
SELECT CustomerName
FROM Customers
WHERE CustomerID IN (
  SELECT CustomerID
  FROM Sales
  WHERE TicketID IN (
    SELECT TicketID
    FROM Tickets
    WHERE TicketType = 'VIP'
  )
);
```

Result Grid

	CustomerName
▶	Surya Rao

Customers 14 x

Output

Action Output

#	Time	Action	Message	Duration / Fetch
✓	26 16:27:25	select * from tickets LIMIT 0, 1000	5 row(s) returned	0.000 sec / 0.000 sec
✓	27 16:34:08	SELECT Customers.CustomerName, Events.EventName, Tickets.TicketType, Tickets.Pr...	5 row(s) returned	0.000 sec / 0.000 sec
✓	28 16:38:50	SELECT CustomerName FROM Customers WHERE CustomerID IN (SELECT Custome...	1 row(s) returned	0.000 sec / 0.000 sec

Object Info Session

Local instance MySQL80

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

dbt
dtcc
eventticketingsystem

Tables

- customers
 - Columns
 - Indexes
 - Foreign Keys
 - Triggers
- events
- reservations
 - Columns
 - Indexes
 - Foreign Keys
 - Triggers
- sales
 - Columns
 - Indexes
 - Foreign Keys
 - Triggers

Administration Schemas

Information

Schema: eventticketingsystem

Object Info Session

File 35* SQL File 36* SQL File 37* SQL File 38* SQL File 39* SQL File 40* SQL File 41* SQL File 42* SQL File 43* SQL File 44* SQL File 45* SQL File 46*

Limit to 1000 rows

```
141 -- Execute the selected portion of the script or everything, if there is no selection
142
143 CREATE VIEW TicketsSalesView AS
144 SELECT Sales.SaleID, Customers.CustomerName, Events.EventName, Tickets.TicketType, Sales.Amount, Sales.SaleDate
145 FROM Sales
146 JOIN Customers ON Sales.CustomerID = Customers.CustomerID
147 JOIN Tickets ON Sales.TicketID = Tickets.TicketID
148 JOIN Events ON Tickets.EventID = Events.EventID;
149
150 -- 4. Using the VIEW: Display all ticket sales
151 SELECT * FROM TicketsSalesView;
152
153 -- 5. STORED PROCEDURE: Count number of tickets sold for an event
154 DELIMITER //
155 CREATE PROCEDURE GetSoldTicketCount(IN inputEventID INT)
156 BEGIN
157     SELECT COUNT(*) AS SoldTickets
158     FROM Tickets
159     WHERE EventID = inputEventID AND Status = 'sold';
160 END //
161 DELIMITER ;
```

Output

Action Output

#	Time	Action	Message	Duration / Fetch
27	16:34:08	SELECT Customers.CustomerName, Events.EventName, Tickets.TicketType, Tickets.Pri...	5 row(s) returned	0.000 sec / 0.000 sec
28	16:38:50	SELECT CustomerName FROM Customers WHERE CustomerID IN (SELECT Custome...	1 row(s) returned	0.000 sec / 0.000 sec
29	16:40:28	CREATE VIEW TicketsSalesView AS SELECT Sales.SaleID, Customers.CustomerName, E...	0 row(s) affected	0.016 sec

Local instance MySQL80

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

dbb

dtcc

eventticketingsystem

Tables

customers

Columns

Indexes

Foreign Keys

Triggers

events

reservations

Columns

Indexes

Foreign Keys

Triggers

sales

Columns

Indexes

Foreign Keys

Triggers

Administration

Schemas

Information

Schema: eventticketingsystem

Object Info

Session

File 35* SQL File 36* SQL File 37* SQL File 38* SQL File 39* SQL File 40* SQL File 41* SQL File 42* SQL File 43* SQL File 44* SQL File 45* SQL File 46*

Limit to 1000 rows

SQL File 46*

140

141

142 -- 3. VIEW: Create a view to see all ticket sales with event names

143 CREATE VIEW TicketSalesView AS

144 SELECT Sales.SaleID, Customers.CustomerName, Events.EventName, Tickets.TicketType, Sales.Amount, Sales.SaleDate

145 FROM Sales

146 JOIN Customers ON Sales.CustomerID = Customers.CustomerID

147 JOIN Tickets ON Sales.TicketID = Tickets.TicketID

148 JOIN Events ON Tickets.EventID = Events.EventID;

149

150 -- 4. Using the VIEW: Display all ticket sales

151 SELECT * FROM TicketSalesView;

152

Result Grid

Filter Rows:

Export:

Wrap Cell Content:

Result Grid

	SaleID	CustomerName	EventName	TicketType	Amount	SaleDate
▶	1	Korkai	Music Concert	Regular	80.00	2025-04-05
	2	Surya Rao	Food Festival	VIP	200.00	2025-04-10
	3	Hari	Food Festival	Regular	100.00	2025-04-12

TicketSalesView 15

Read Only

Output

Action Output

#	Time	Action	Message	Duration / Fetch
28	16:38:50	SELECT CustomerName FROM Customers WHERE CustomerID IN (SELECT Custome...	1 row(s) returned	0.000 sec / 0.000 sec
29	16:40:28	CREATE VIEW TicketSalesView AS SELECT Sales.SaleID, Customers.CustomerName, E...	0 row(s) affected	0.015 sec
30	16:41:55	SELECT * FROM TicketSalesView LIMIT 0, 1000	3 row(s) returned	0.000 sec / 0.000 sec

Local instance MySQL80 x

File Edit View Query Database Server Tools Scripting Help

Navigator: .File 35* SQL File 36* SQL File 37* SQL File 38* SQL File 39* SQL File 40* SQL File 41* SQL File 42* SQL File 43* SQL File 44* SQL File 45* SQL File 45*

SCHEMAS

Filter objects

- dtb
- dtcc
- eventticketingsystem**
 - Tables
 - customers
 - Columns
 - Indexes
 - Foreign Keys
 - Triggers
 - events
 - Columns
 - Indexes
 - Foreign Keys
 - Triggers
 - reservations
 - Columns
 - Indexes
 - Foreign Keys
 - Triggers
 - sales
 - Columns
 - Indexes
 - Foreign Keys
 - Triggers
- Administration
- Schemas

Information

Schema: eventticketingsystem

```
151 SELECT * FROM TicketSalesView;
152
153 -- 5. STORED PROCEDURE: Count number of tickets sold for an event
154 DELIMITER //
155 CREATE PROCEDURE GetSoldTicketCount(IN inputEventID INT)
156 BEGIN
157     SELECT COUNT(*) AS SoldTickets
158     FROM Tickets
159     WHERE EventID = inputEventID AND Status = 'Sold';
160 END //
161 DELIMITER ;
162
163 -- Execute Stored Procedure Example
164 CALL GetSoldTicketCount(1);
165
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: |

SoldTickets	
1	

Result 16 x | Read Only

Output

Action Output

#	Time	Action	Message	Duration / Fetch
30	16:41:55	SELECT * FROM TicketSalesView LIMIT 0, 1000	3 row(s) returned	0.000 sec / 0.000 sec
31	16:51:41	CREATE PROCEDURE GetSoldTicketCount(IN inputEventID INT) BEGIN SELECT C...	0 row(s) affected	0.016 sec
32	16:51:41	-- Execute Stored Procedure Example CALL GetSoldTicketCount(1)	1 row(s) returned	0.000 sec / 0.000 sec

Object Info Session

Search

ENG

16:51