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
- frnd
- frnk
- games
- hotel
- hotell
- india
- 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*

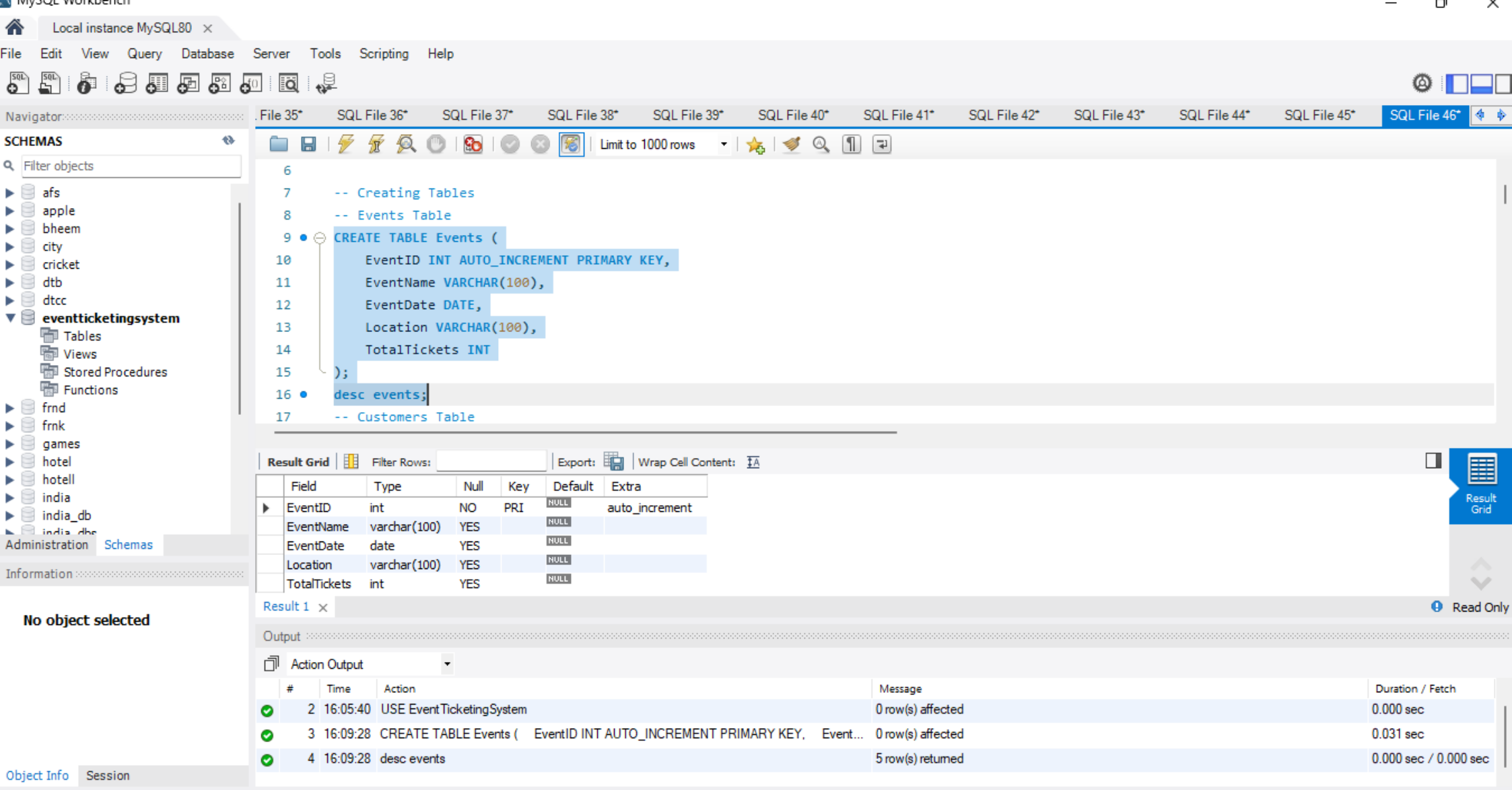
Limit to 1000 rows

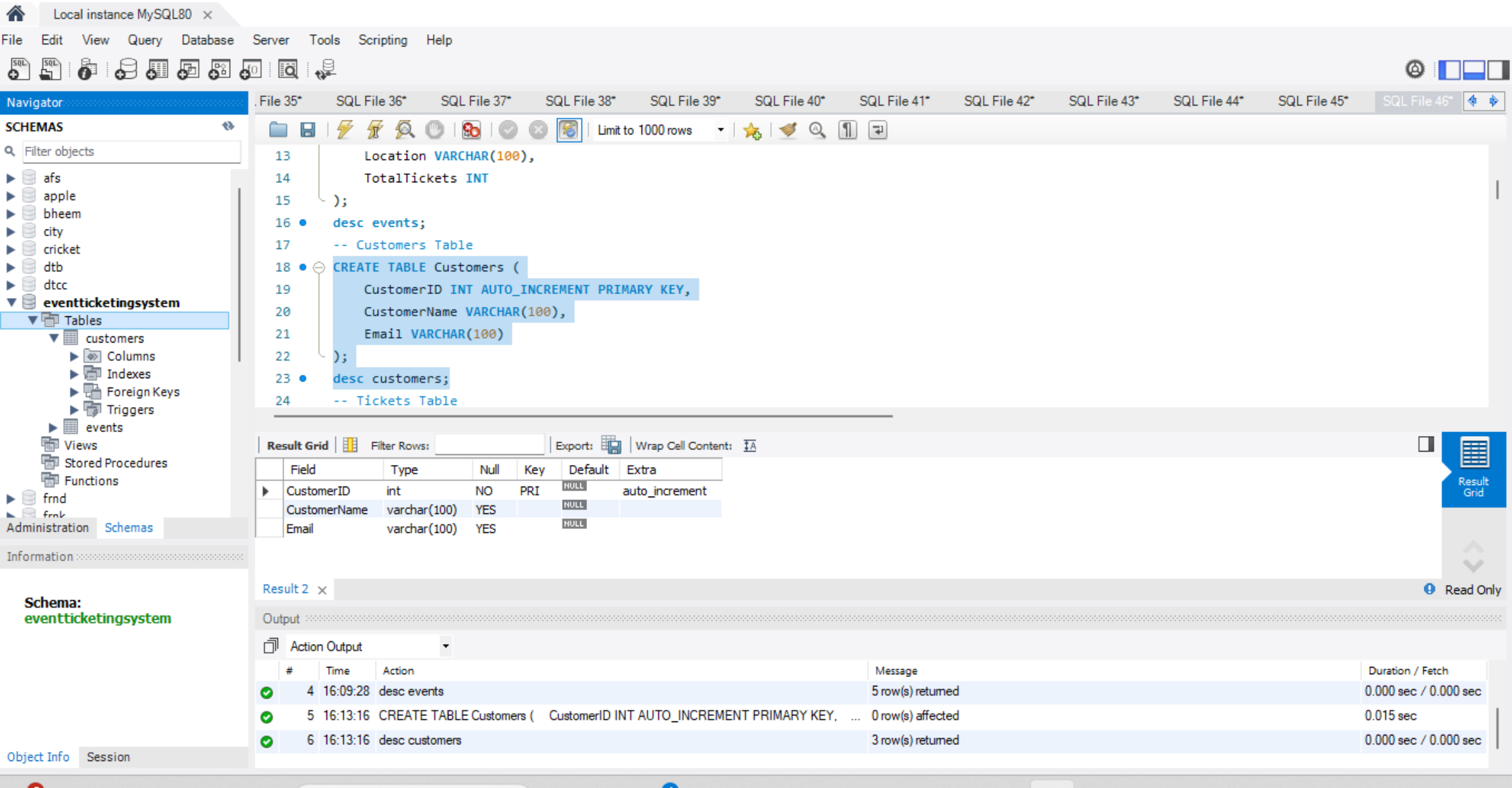
```
1
2   -- Create Database
3
4   CREATE DATABASE EventTicketingSystem;
5   USE EventTicketingSystem;
6
7   -- Create Tables
8
9
10  -- Events Table
11  CREATE TABLE Events (
12      EventID INT AUTO_INCREMENT PRIMARY KEY,
13      EventName VARCHAR(100),
14      EventDate DATE,
15      Location VARCHAR(100),
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
SCHEMAS
Filter objects
afs
apple
bheem
city
cricket
dtb
dtcc
eventticketingsystem
Tables
Views
Stored Procedures
Functions
frnd
frnk
games
hotel
hotell
india
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 Filter Rows: Export: Wrap Cell Content:
Field Type Null 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
Read Only
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 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
Views
Stored Procedures
Functions
frnd
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
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 Filter Rows: Export: Wrap Cell Content:
Field Type Null Key Default Extra
CustomerID int NO PRI NULL auto_increment
CustomerName varchar(100) YES NULL
Email varchar(100) YES NULL
Result Grid
Result 2
Read Only
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

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

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

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

Result Grid

Result 3

Read Only

Output

Action Output

#	Time	Action	Message	Duration / Fetch
6	16:13:16	desc customers	3 row(s) returned	0.000 sec / 0.000 sec
7	16:15:49	CREATE TABLE Tickets (TicketID INT AUTO_INCREMENT PRIMARY KEY, Even...	0 row(s) affected	0.032 sec
8	16:15:49	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
- dtb
- dtcc
- **eventticketingsystem**
  - Tables
    - customers
      - Columns
      - Indexes
      - Foreign Keys
      - Triggers
    - events
    - reservations
      - 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

```
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 | Filter Rows: | Export: | Wrap Cell Content:

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 Grid

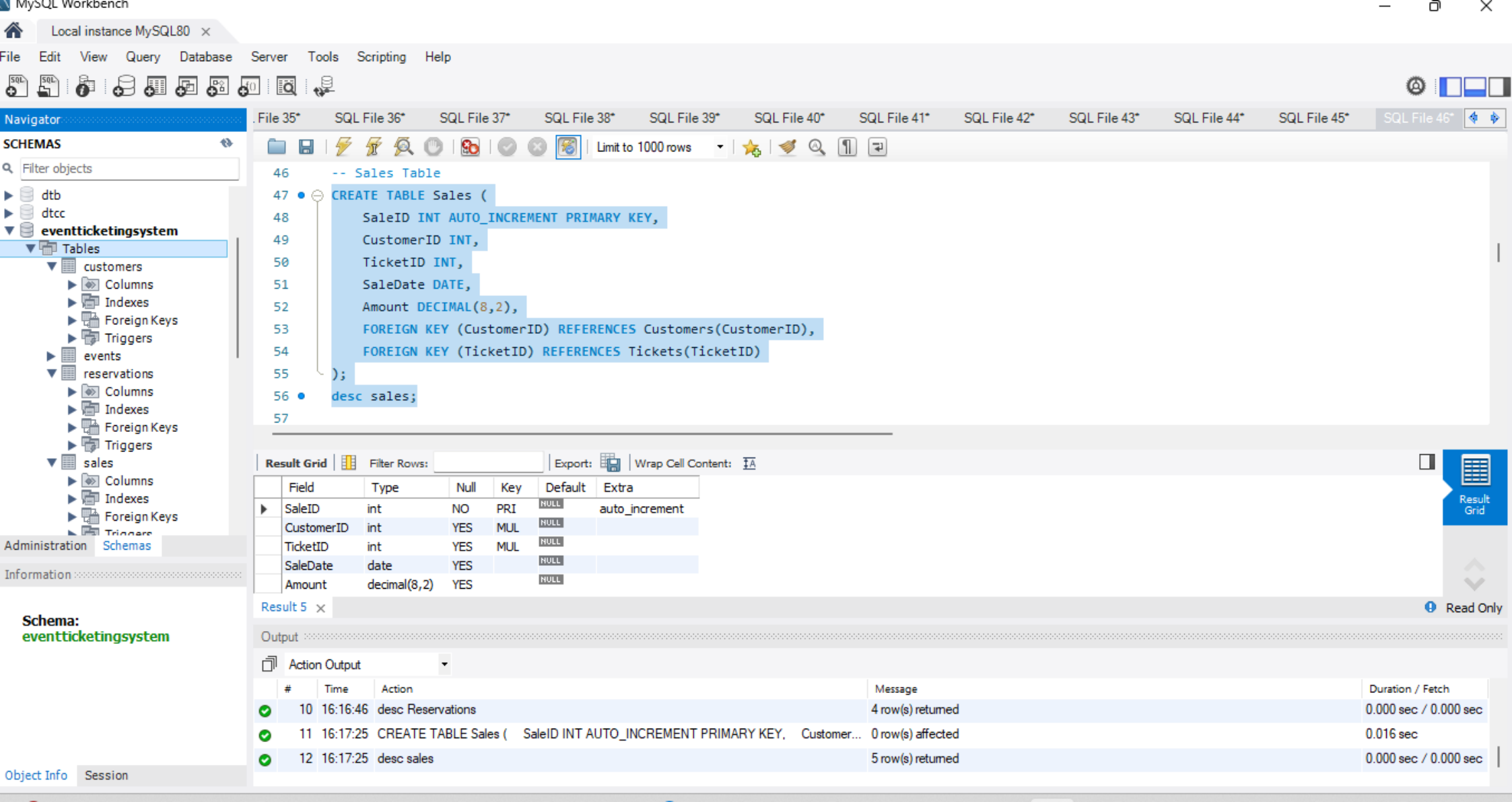
Result 4

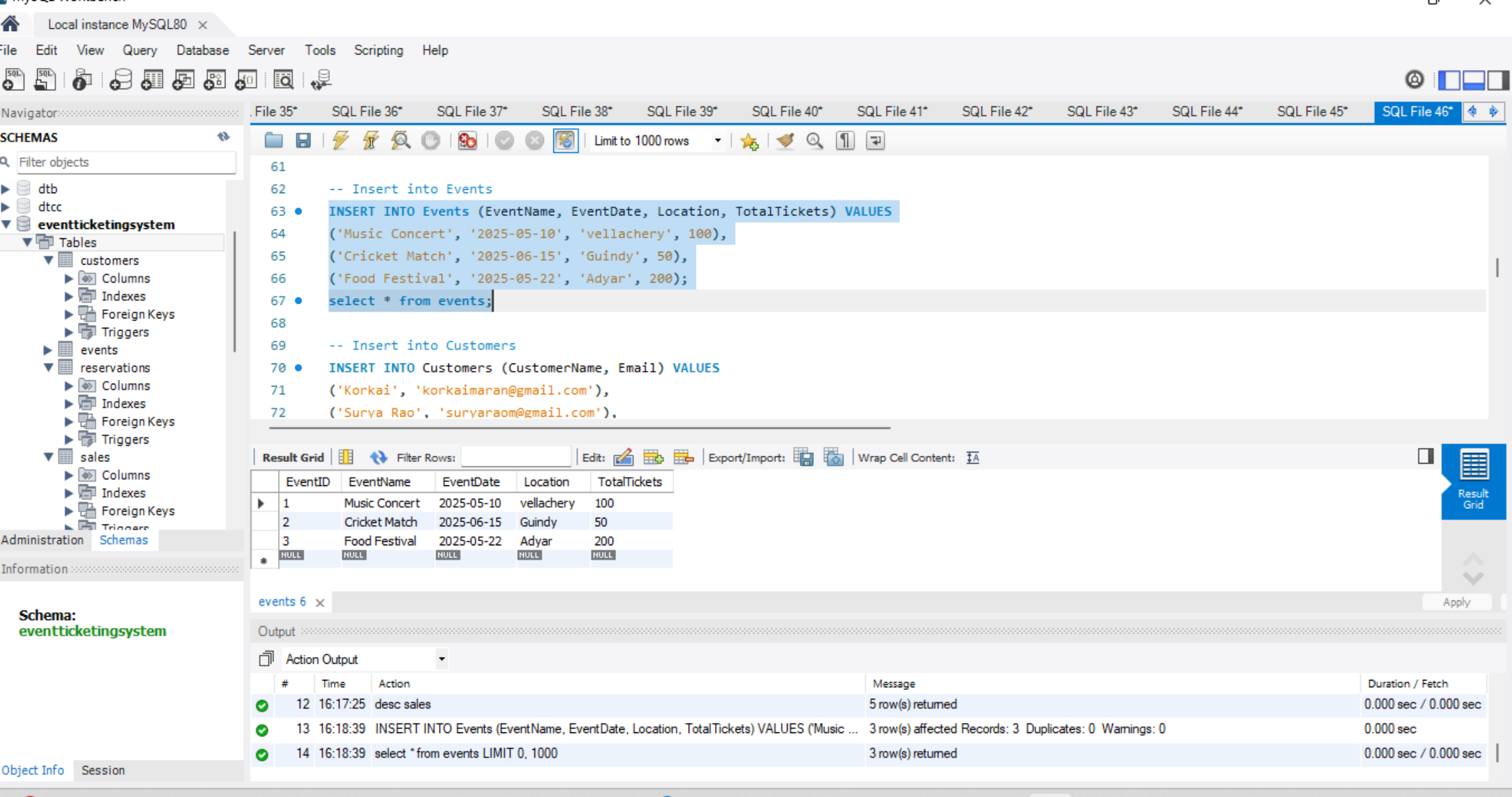
Read Only

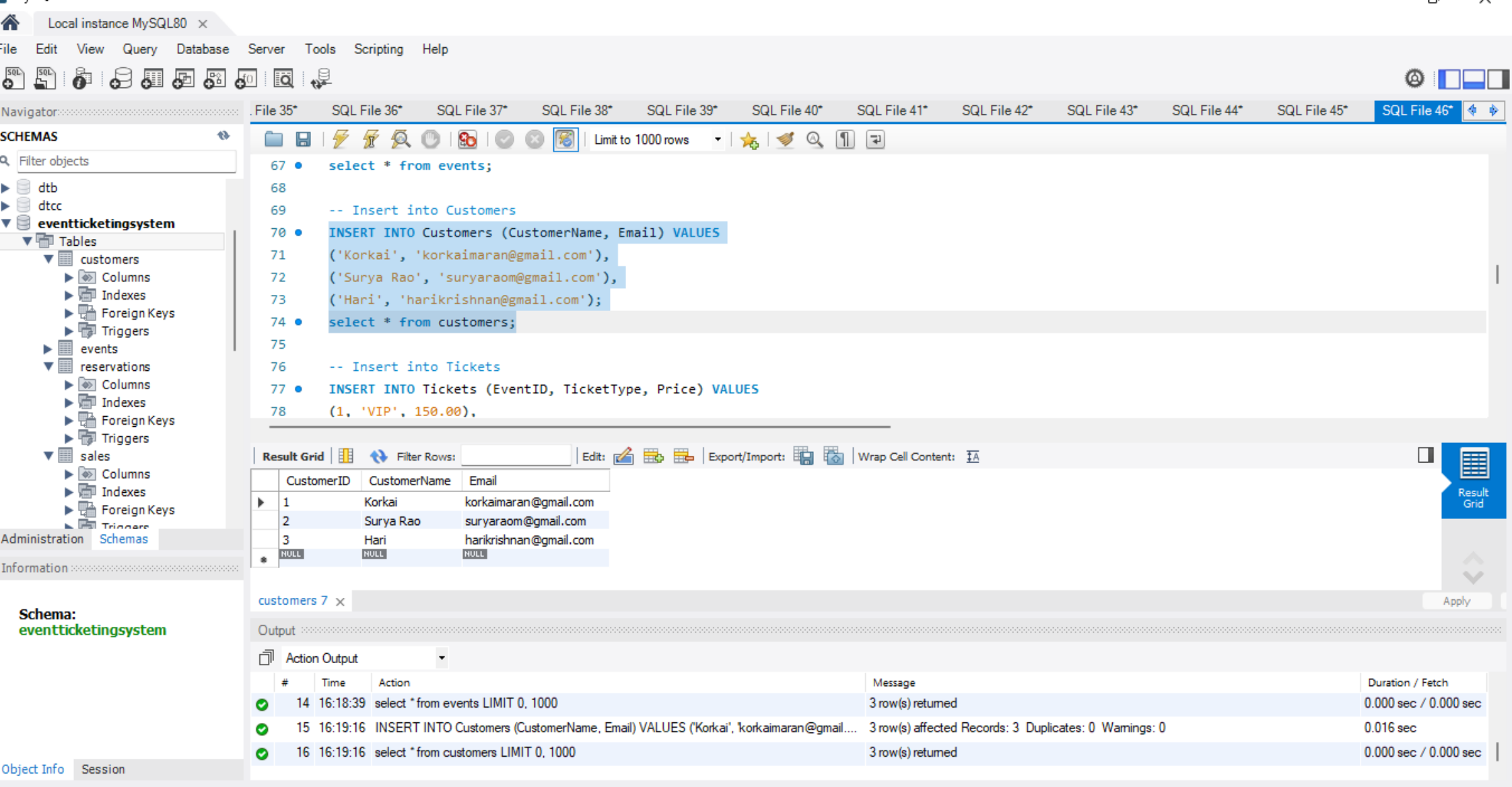
Output

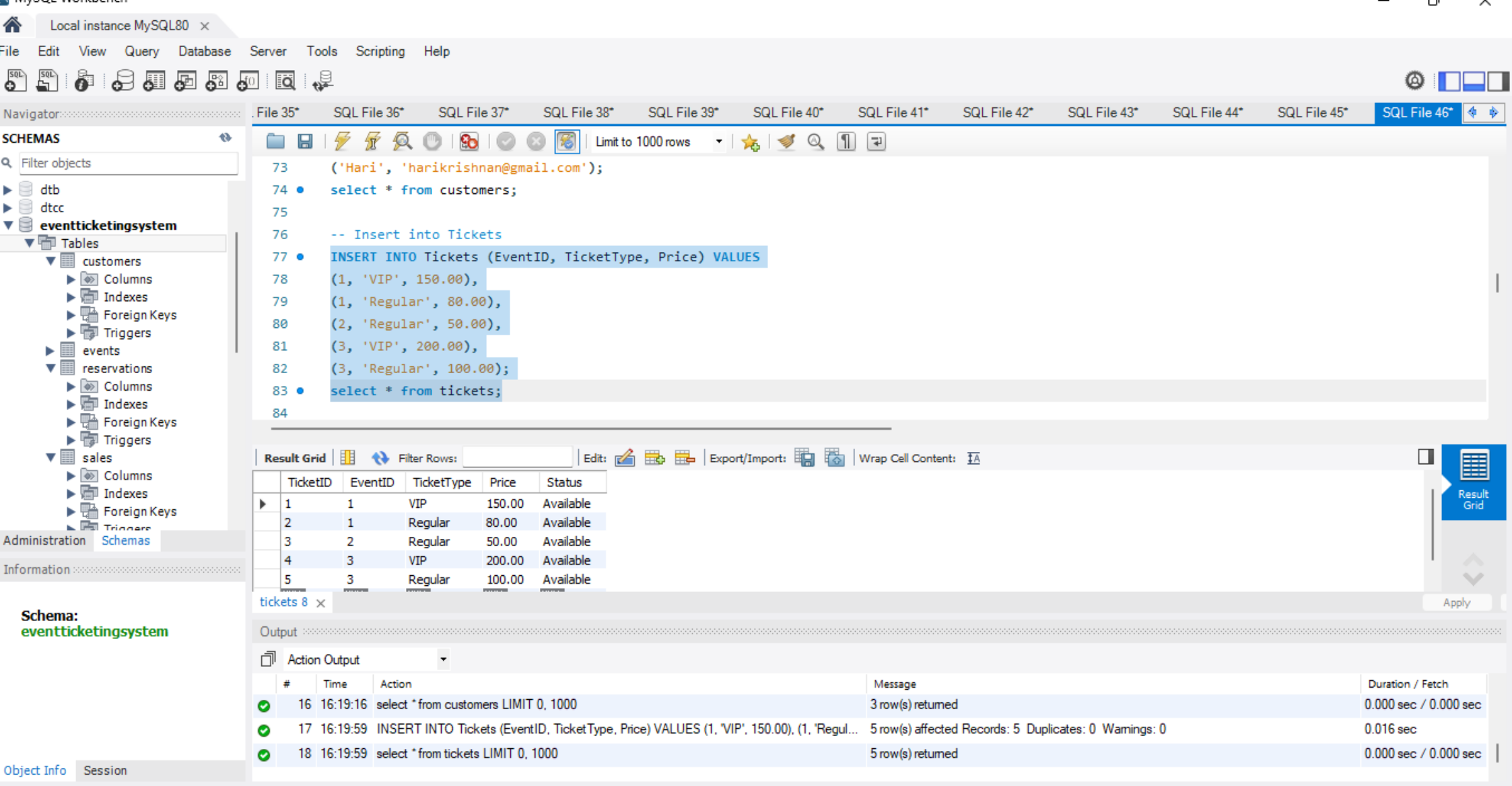
Action Output

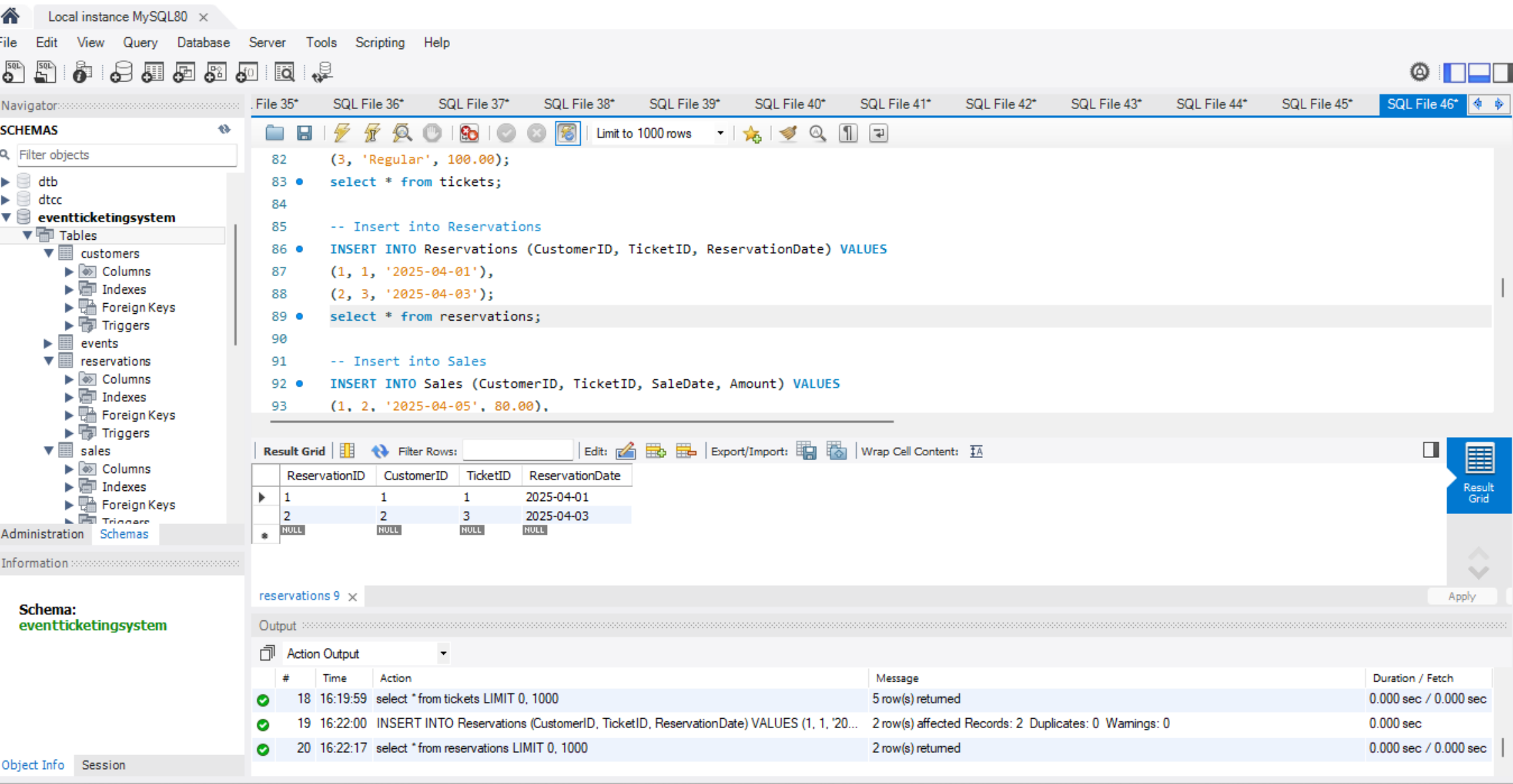
#	Time	Action	Message	Duration / Fetch
8	16:15:49	desc tickets	5 row(s) returned	0.015 sec / 0.000 sec
9	16:16:46	CREATE TABLE Reservations (ReservationID INT AUTO_INCREMENT PRIMARY K...	0 row(s) affected	0.031 sec
10	16:16:46	desc Reservations	4 row(s) returned	0.000 sec / 0.000 sec

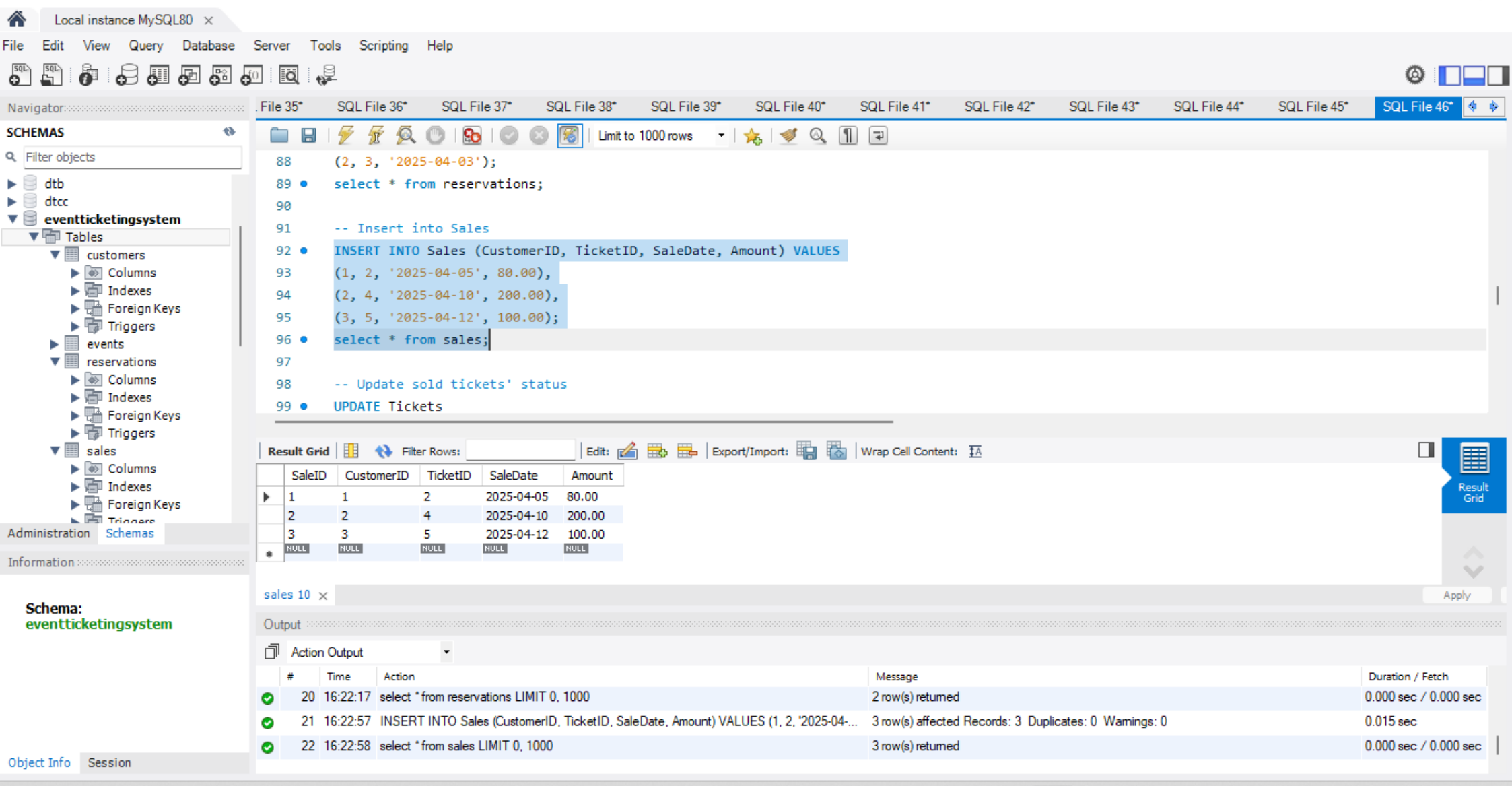

MySQL Workbench
Local instance MySQL80
File Edit View Query Database Server Tools Scripting Help
Navigator
SCHEMAS
Filter objects
dtb
dtcc
eventticketingsystem
Tables
customers
Columns
Indexes
Foreign Keys
Triggers
events
reservations
Columns
Indexes
Foreign Keys
Triggers
sales
Columns
Indexes
Foreign Keys
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
46 -- Sales Table
47 CREATE TABLE Sales (
48 SaleID INT AUTO_INCREMENT PRIMARY KEY,
49 CustomerID INT,
50 TicketID INT,
51 SaleDate DATE,
52 Amount DECIMAL(8,2),
53 FOREIGN KEY (CustomerID) REFERENCES Customers(CustomerID),
54 FOREIGN KEY (TicketID) REFERENCES Tickets(TicketID)
55);
56 desc sales;
57
Result Grid Filter Rows: Export: Wrap Cell Content:
Field Type Null Key Default Extra
SaleID int NO PRI NULL auto_increment
CustomerID int YES MUL NULL
TicketID int YES MUL NULL
SaleDate date YES NULL
Amount decimal(8,2) YES NULL
Result Grid
Result 5
Read Only
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

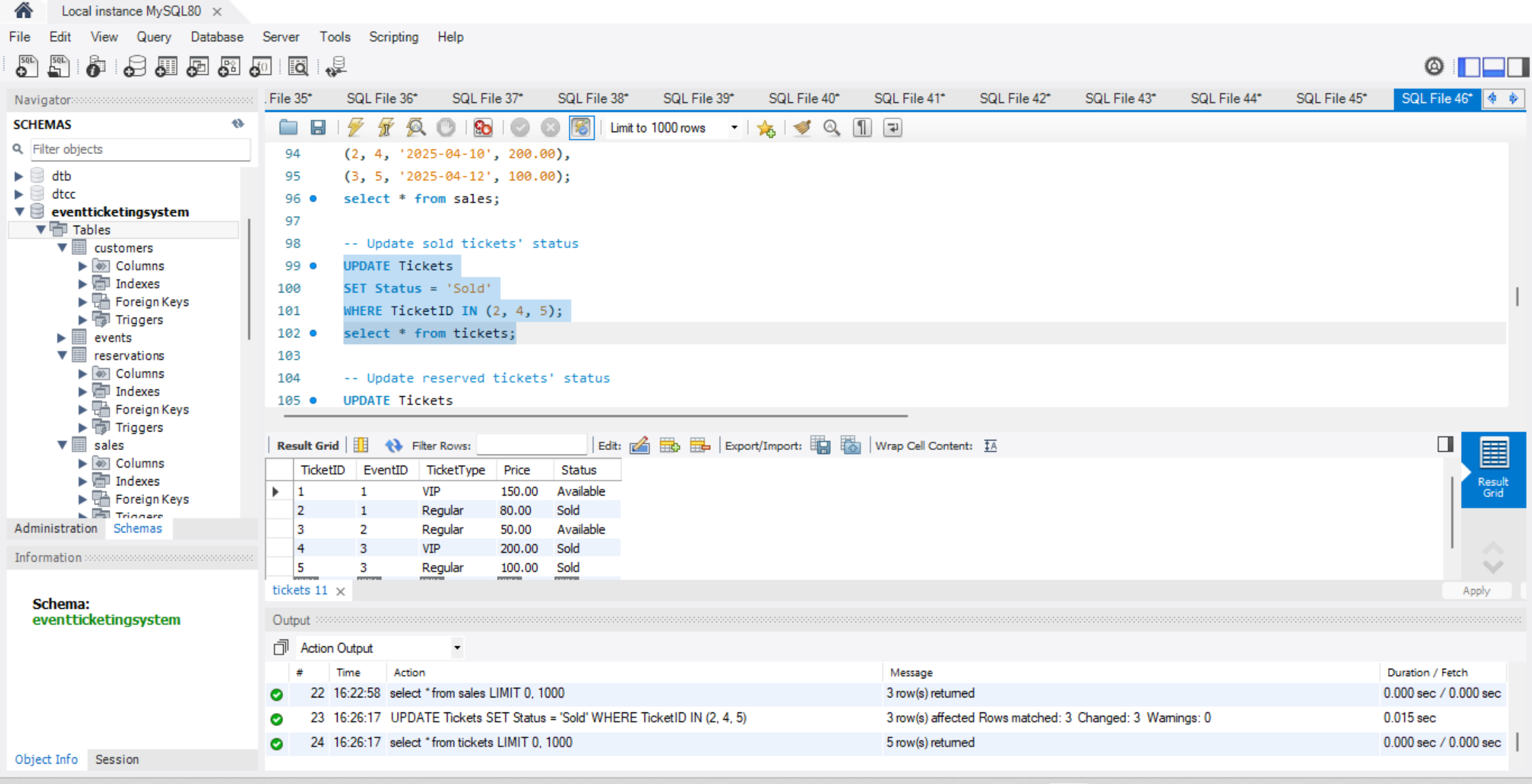

Local instance MySQL80
File Edit View Query Database Server Tools Scripting Help
Navigator
SCHEMAS
Filter objects
dtb
dtcc
eventticketingsystem
Tables
customers
Columns
Indexes
Foreign Keys
Triggers
events
reservations
Columns
Indexes
Foreign Keys
Triggers
sales
Columns
Indexes
Foreign Keys
Administration 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', 'Guindy', 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', 'suryaraom@gmail.com'),
Result Grid Filter Rows: Edit: Export/Import: Wrap Cell Content:
EventID EventName EventDate Location TotalTickets
1 Music Concert 2025-05-10 vellachery 100
2 Cricket Match 2025-06-15 Guindy 50
3 Food Festival 2025-05-22 Adyar 200
NULL NULL NULL NULL NULL
Result Grid
events 6
Apply
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

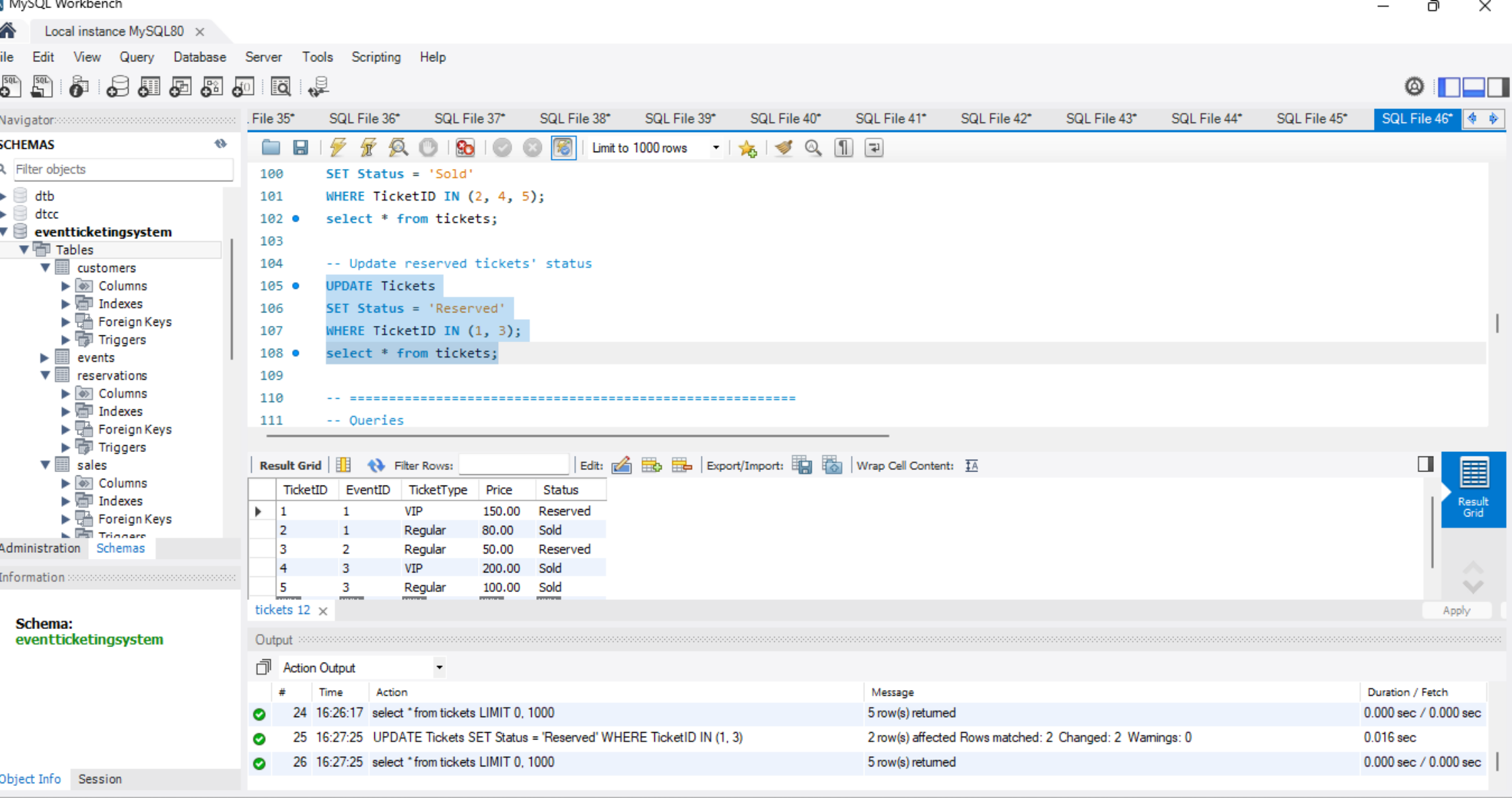

Local instance MySQL80
File Edit View Query Database Server Tools Scripting Help
Navigator
SCHEMAS
Filter objects
dtb
dtcc
eventticketingsystem
Tables
customers
Columns
Indexes
Foreign Keys
Triggers
events
reservations
Columns
Indexes
Foreign Keys
Triggers
sales
Columns
Indexes
Foreign Keys
Administration 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
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 Filter Rows: Edit: Export/Import: Wrap Cell Content:
CustomerID CustomerName Email
1 Korkai korkaimaran@gmail.com
2 Surya Rao suryaraom@gmail.com
3 Hari harikrishnan@gmail.com
NULL NULL NULL
Result Grid
customers 7
Apply
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.... 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

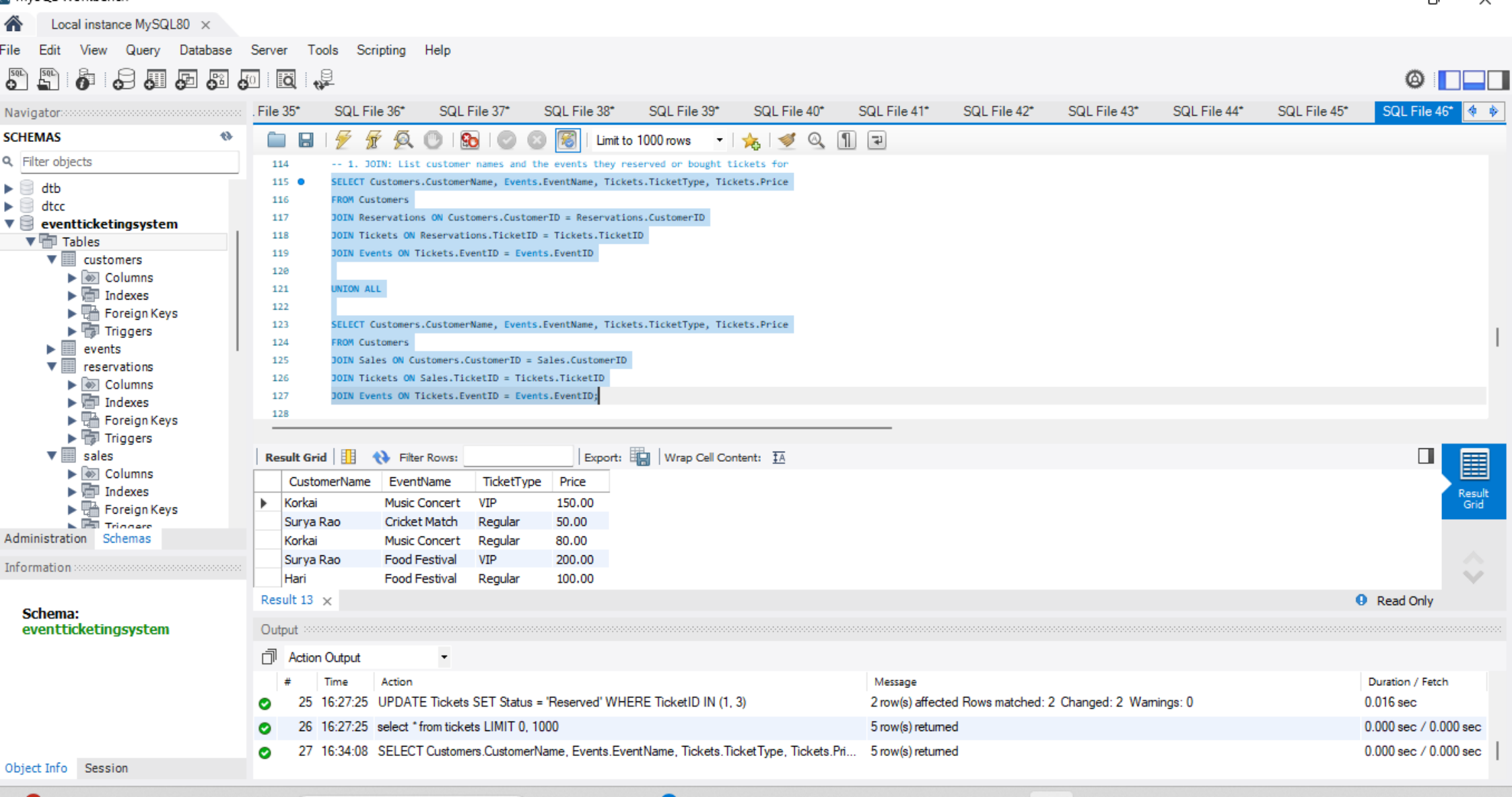

Local instance MySQL80
File Edit View Query Database Server Tools Scripting Help
Navigator
SCHEMAS
Filter objects
dtb
dtcc
eventticketingsystem
Tables
customers
Columns
Indexes
Foreign Keys
Triggers
events
reservations
Columns
Indexes
Foreign Keys
Triggers
sales
Columns
Indexes
Foreign Keys
Administration 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
('Hari', 'harikrishnan@gmail.com');
select * from customers;
-- Insert into Tickets
INSERT INTO Tickets (EventID, TicketType, Price) VALUES
(1, 'VIP', 150.00),
(1, 'Regular', 80.00),
(2, 'Regular', 50.00),
(3, 'VIP', 200.00),
(3, 'Regular', 100.00);
select * from tickets;
Result Grid Filter Rows: Edit: Export/Import: Wrap Cell Content:
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 8
Result Grid
Apply
Output
Action Output
Time Action Message Duration / Fetch
16 16:19:16 select * from customers LIMIT 0, 1000 3 row(s) returned 0.000 sec / 0.000 sec
17 16:19:59 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 16:19:59 select * from tickets LIMIT 0, 1000 5 row(s) returned 0.000 sec / 0.000 sec

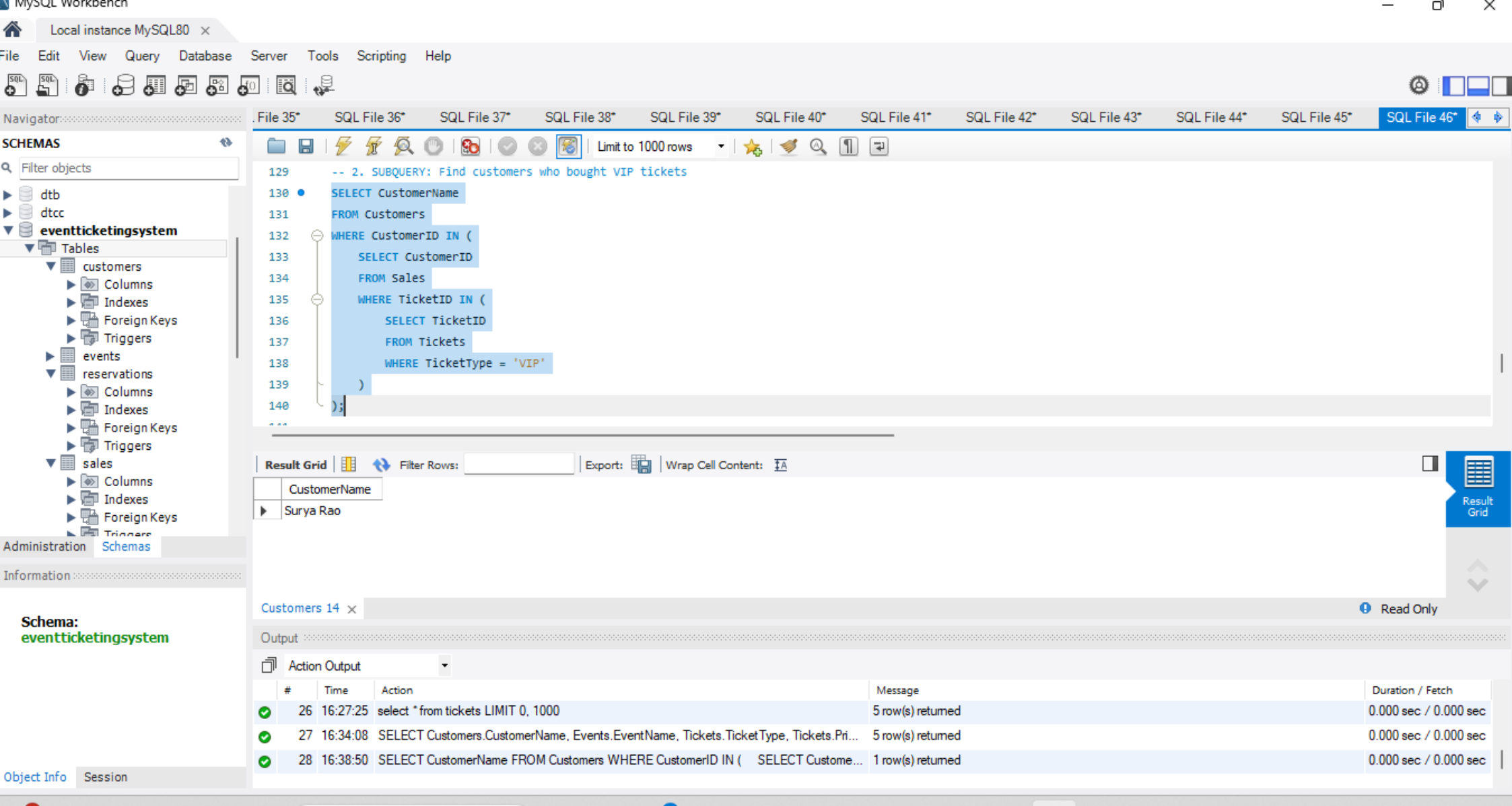

Local instance MySQL80
File Edit View Query Database Server Tools Scripting Help
Navigator
SCHEMAS
Filter objects
dtb
dtcc
eventticketingsystem
Tables
customers
Columns
Indexes
Foreign Keys
Triggers
events
reservations
Columns
Indexes
Foreign Keys
Triggers
sales
Columns
Indexes
Foreign Keys
Administration 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
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: Edit: 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
Result Grid
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

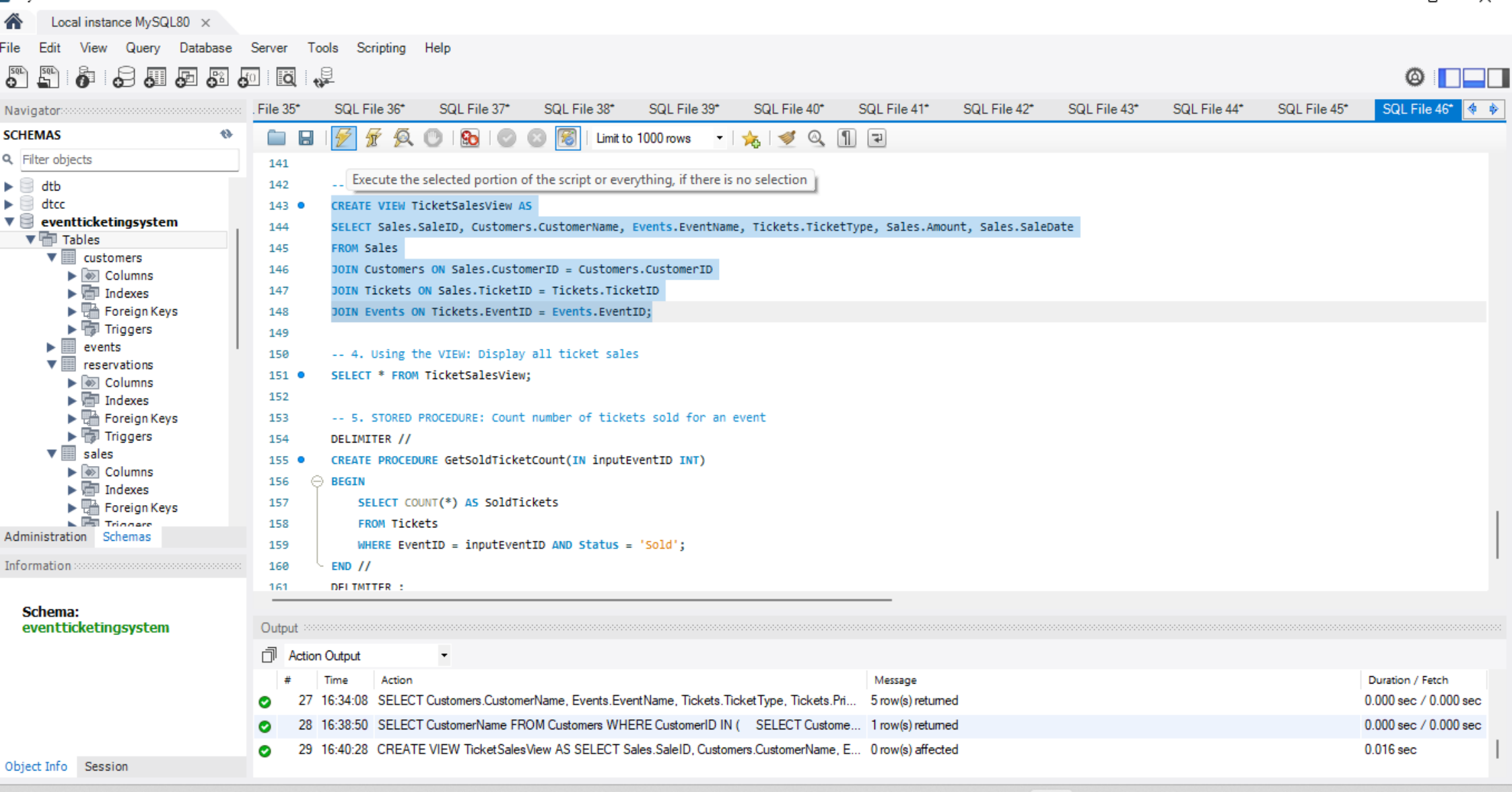

Local instance MySQL80
File Edit View Query Database Server Tools Scripting Help
Navigator
SCHEMAS
Filter objects
dtb
dtcc
eventticketingsystem
Tables
customers
Columns
Indexes
Foreign Keys
Triggers
events
reservations
Columns
Indexes
Foreign Keys
Triggers
sales
Columns
Indexes
Foreign Keys
Administration 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 Filter Rows: Edit: Export/Import: Wrap Cell Content:
SaleID CustomerID TicketID SaleDate Amount
1 1 2 2025-04-05 80.00
2 2 4 2025-04-10 200.00
3 3 5 2025-04-12 100.00
NULL NULL NULL NULL NULL
Result Grid
sales 10
Apply
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-... 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

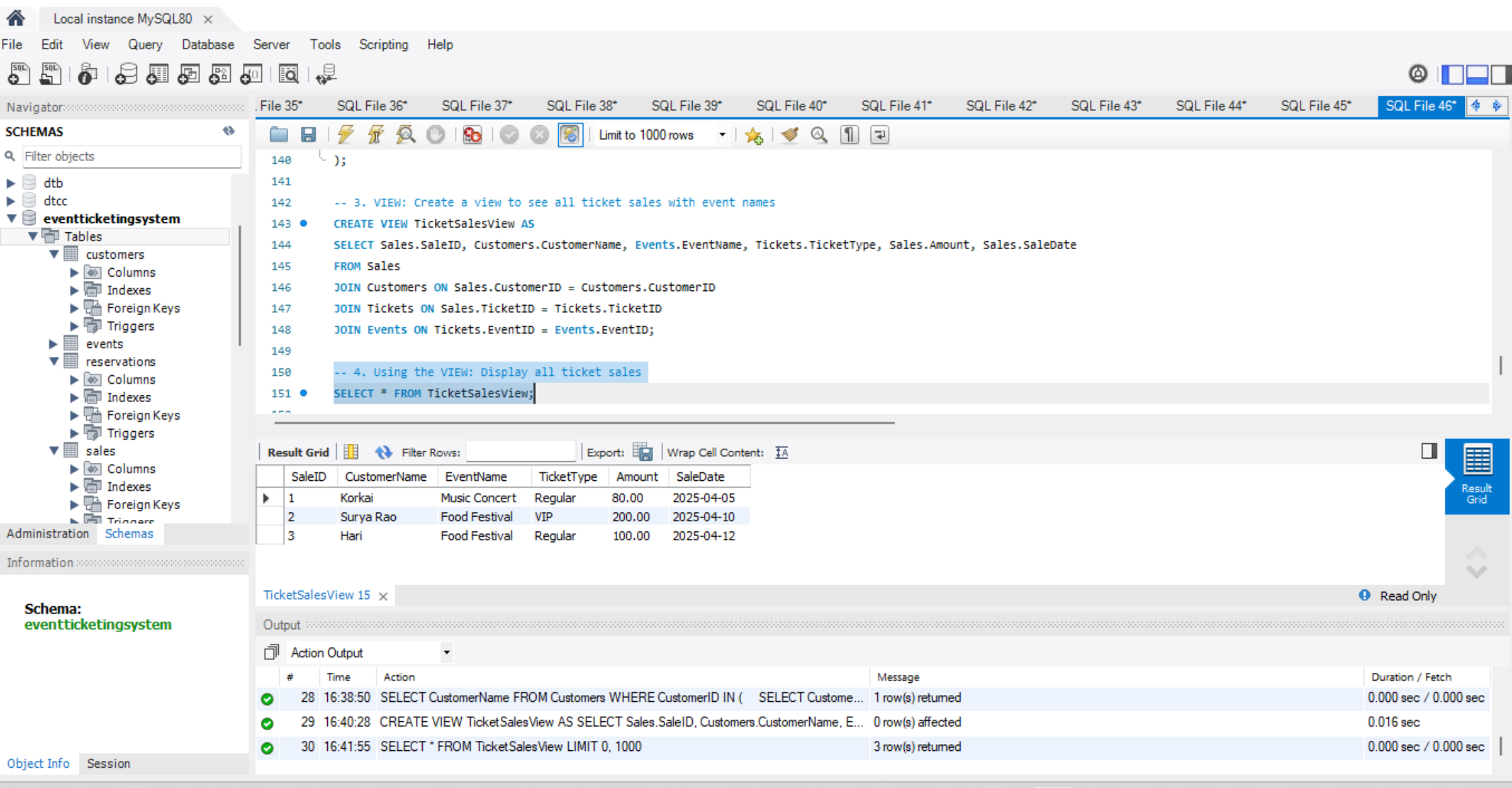

Local instance MySQL80
File Edit View Query Database Server Tools Scripting Help
Navigator
SCHEMAS
Filter objects
dtb
dtcc
eventticketingsystem
Tables
customers
Columns
Indexes
Foreign Keys
Triggers
events
reservations
Columns
Indexes
Foreign Keys
Triggers
sales
Columns
Indexes
Foreign Keys
Administration 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 Filter Rows: Edit: Export/Import: Wrap Cell Content:
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 11
Result Grid
Apply
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
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
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 Filter Rows: Edit: Export/Import: Wrap Cell Content:
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
Result Grid
Apply
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
dtb
dtcc
eventticketingsystem
Tables
customers
Columns
Indexes
Foreign Keys
Triggers
events
reservations
Columns
Indexes
Foreign Keys
Triggers
sales
Columns
Indexes
Foreign Keys
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
114 -- 1. JOIN: List customer names and the events they reserved or bought tickets for
115 SELECT Customers.CustomerName, Events.EventName, Tickets.TicketType, Tickets.Price
116 FROM Customers
117 JOIN Reservations ON Customers.CustomerID = Reservations.CustomerID
118 JOIN Tickets ON Reservations.TicketID = Tickets.TicketID
119 JOIN Events ON Tickets.EventID = Events.EventID
120
121 UNION ALL
122
123 SELECT Customers.CustomerName, Events.EventName, Tickets.TicketType, Tickets.Price
124 FROM Customers
125 JOIN Sales ON Customers.CustomerID = Sales.CustomerID
126 JOIN Tickets ON Sales.TicketID = Tickets.TicketID
127 JOIN Events ON Tickets.EventID = Events.EventID;
128
Result Grid Filter Rows: Export: Wrap Cell Content:
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
Read Only
Result Grid
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.Pri... 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
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
129 -- 2. SUBQUERY: Find customers who bought VIP tickets
130 SELECT CustomerName
131 FROM Customers
132 WHERE CustomerID IN (
133 SELECT CustomerID
134 FROM Sales
135 WHERE TicketID IN (
136 SELECT TicketID
137 FROM Tickets
138 WHERE TicketType = 'VIP'
139)
140);
Result Grid Filter Rows: Export: Wrap Cell Content:
CustomerName
Surya Rao
Result Grid
Customers 14
Read Only
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.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


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
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
Execute the selected portion of the script or everything, if there is no selection
141
142 --
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
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 TicketSalesView 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
dtb
dtcc
eventticketingsystem
Tables
customers
Columns
Indexes
Foreign Keys
Triggers
events
reservations
Columns
Indexes
Foreign Keys
Triggers
sales
Columns
Indexes
Foreign Keys
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
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;
Result Grid Filter Rows: Export: Wrap Cell Content:
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
Result Grid
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 0.016 sec
30 16:41:55 SELECT * FROM TicketSalesView 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

```
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 Grid

Result 16 | 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

Search

ENG

16:51