MySQL Workbench

Local instance MySQL80

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

- collage
- empinfo
  - Tables
  - Views
  - Stored Procedures
  - Functions
- kirandb
- librarydb
  - Tables
    - books
    - fines
    - loans
    - reservations
    - users
  - Views
  - Stored Procedures
  - Functions

Administration Schemas

Information

No object selected

Object Info Session

SQL File 1* SQL File 3* SQL File 4* SQL File 5* SQL File 6* SQL File 7* SQL File 8* SQL File 9* LibraryManagementdb

Limit to 1000 rows

```sql
create database LibraryDB;
use LibraryDB;

CREATE TABLE Users (
    user_id INT PRIMARY KEY AUTO_INCREMENT,
    name VARCHAR(100) NOT NULL,
    email VARCHAR(100) UNIQUE NOT NULL,
    phone VARCHAR(15),
    address TEXT,
    membership_type ENUM('Student', 'Faculty', 'Guest') NOT NULL,
    created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP
);

CREATE TABLE Books (
    book_id INT PRIMARY KEY AUTO_INCREMENT,
    title VARCHAR(255) NOT NULL,
    author VARCHAR(100) NOT NULL,
    publisher VARCHAR(100),
    published_year INT,
    isbn VARCHAR(20) UNIQUE NOT NULL,
    category VARCHAR(100),
    copies_available INT DEFAULT 1
);
```

1.CREATING DATABASE AND USING IT

Output

Action Output

#	Time	Action	Message	Duration / Fetch
1	17:59:14	use empinfo	0 row(s) affected	0.000 sec

36°C Mostly sunny

Search

ENG IN

06:02 PM 18-03-2025

MySQL Workbench

Local instance MySQL80

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

- collage
- empinfo
  - Tables
  - Views
  - Stored Procedures
  - Functions
- kirandb
- librarydb
  - Tables
    - books
    - fines
    - loans
    - reservations
    - users
  - Views
  - Stored Procedures
  - Functions

Administration Schemas

Information

No object selected

Object Info Session

SQL File 1* SQL File 3* SQL File 4* SQL File 5* SQL File 6* SQL File 7* SQL File 8* SQL File 9* LibraryManagementdb

Limit to 1000 rows

2.CREATING USER TABLE

```sql
create database LibraryDB;
use LibraryDB;

CREATE TABLE Users (
    user_id INT PRIMARY KEY AUTO_INCREMENT,
    name VARCHAR(100) NOT NULL,
    email VARCHAR(100) UNIQUE NOT NULL,
    phone VARCHAR(15),
    address TEXT,
    membership_type ENUM('Student', 'Faculty', 'Guest') NOT NULL,
    created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP
);

CREATE TABLE Books (
    book_id INT PRIMARY KEY AUTO_INCREMENT,
    title VARCHAR(255) NOT NULL,
    author VARCHAR(100) NOT NULL,
    publisher VARCHAR(100),
    published_year INT,
    isbn VARCHAR(20) UNIQUE NOT NULL,
    category VARCHAR(100),
    copies_available INT DEFAULT 1
);
```

Output

Action Output

#	Time	Action	Message	Duration / Fetch
1	17:59:14	use empinfo	0 row(s) affected	0.000 sec

36°C Mostly sunny

Search

ENG IN

06:03 PM 18-03-2025

MySQL Workbench

Local instance MySQL80

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

- collage
- empinfo
  - Tables
  - Views
  - Stored Procedures
  - Functions
- kirandb
- librarydb
  - Tables
    - books
    - fines
    - loans
    - reservations
    - users
  - Views
  - Stored Procedures
  - Functions

Administration Schemas

Information

No object selected

Object Info Session

SQL File 1* SQL File 3* SQL File 4* SQL File 5* SQL File 6* SQL File 7* SQL File 8* SQL File 9* LibraryManagementdb

Limit to 1000 rows

3.CREATING BOOKS TABLE

```
use LibraryDB;

CREATE TABLE Users (
    user_id INT PRIMARY KEY AUTO_INCREMENT,
    name VARCHAR(100) NOT NULL,
    email VARCHAR(100) UNIQUE NOT NULL,
    phone VARCHAR(15),
    address TEXT,
    membership_type ENUM('Student', 'Faculty', 'Guest') NOT NULL,
    created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP
);

CREATE TABLE Books (
    book_id INT PRIMARY KEY AUTO_INCREMENT,
    title VARCHAR(255) NOT NULL,
    author VARCHAR(100) NOT NULL,
    publisher VARCHAR(100),
    published_year INT,
    isbn VARCHAR(20) UNIQUE NOT NULL,
    category VARCHAR(100),
    copies_available INT DEFAULT 1
);
```

Output

Action Output

#	Time	Action	Message	Duration / Fetch
1	17:59:14	use empinfo	0 row(s) affected	0.000 sec

36°C Mostly sunny

Search

ENG IN

06:03 PM 18-03-2025

MySQL Workbench

Local instance MySQL80

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

- collage
- empinfo
  - Tables
  - Views
  - Stored Procedures
  - Functions
- kirandb
- librarydb
  - Tables
    - books
    - fines
    - loans
    - reservations
    - users
  - Views
  - Stored Procedures
  - Functions

Administration Schemas

Information

No object selected

Object Info Session

SQL File 1* SQL File 3* SQL File 4* SQL File 5* SQL File 6* SQL File 7* SQL File 8* SQL File 9* LibraryManagementdb

Limit to 1000 rows

```
29      loan_date TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
30      due_date DATE NOT NULL,
31      return_date DATE NULL,
32      status ENUM('Borrowed', 'Returned', 'Overdue') DEFAULT 'Borrowed',
33      FOREIGN KEY (user_id) REFERENCES Users(user_id) ON DELETE CASCADE,
34      FOREIGN KEY (book_id) REFERENCES Books(book_id) ON DELETE CASCADE
35  );
36
37  CREATE TABLE Reservations (
38      reservation_id INT AUTO_INCREMENT PRIMARY KEY,
39      user_id INT NOT NULL,
40      book_id INT NOT NULL,
41      reservation_date TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
42      status ENUM('Pending', 'Completed', 'Cancelled') DEFAULT 'Pending',
43      FOREIGN KEY (user_id) REFERENCES Users(user_id),
44      FOREIGN KEY (book_id) REFERENCES Books(book_id)
45  );
46
47
48  CREATE TABLE Fines (
49      fine_id INT PRIMARY KEY AUTO_INCREMENT,
50      user_id INT,
51      loan id INT,
```

4.CREATING RESERVATION TABLE

Output

Action Output

#	Time	Action	Message	Duration / Fetch
1	17:59:14	use empinfo	0 row(s) affected	0.000 sec

36°C Mostly sunny

Search

ENG IN

06:04 PM 18-03-2025

MySQL Workbench

Local instance MySQL80

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

- collage
- empinfo
  - Tables
  - Views
  - Stored Procedures
  - Functions
- kirandb
- librarydb
  - Tables
    - books
    - fines
    - loans
    - reservations
    - users
  - Views
  - Stored Procedures
  - Functions

Administration Schemas

Information

No object selected

Object Info Session

SQL File 1* SQL File 3* SQL File 4* SQL File 5* SQL File 6* SQL File 7* SQL File 8* SQL File 9* LibraryManagementdb

Limit to 1000 rows

```
41      reservation_date TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
42      status ENUM('Pending', 'Completed', 'Cancelled') DEFAULT 'Pending',
43      FOREIGN KEY (user_id) REFERENCES Users(user_id),
44      FOREIGN KEY (book_id) REFERENCES Books(book_id)
45  );
46
47
48  CREATE TABLE Fines (
49      fine_id INT PRIMARY KEY AUTO_INCREMENT,
50      user_id INT,
51      loan_id INT,
52      amount DECIMAL(5,2) NOT NULL,
53      paid BOOLEAN DEFAULT FALSE,
54      FOREIGN KEY (user_id) REFERENCES Users(user_id) ON DELETE CASCADE,
55      FOREIGN KEY (loan_id) REFERENCES Loans(loan_id) ON DELETE CASCADE
56  );
57
58  -- Adding Users
59  INSERT INTO Users (name, email, phone, address, membership_type)
60  VALUES
61  ('Siva', 'siva@example.com', '1234567890', '123 Main St', 'Student'),
62  ('Ram', 'ram@example.com', '9876543210', '456 Elm St', 'Faculty'),
63  ('John', 'john@example.com', '1122334455', '789 Oak St', 'Guest'),
```

5.CREATING FINES TABLE

Output

Action Output

#	Time	Action	Message	Duration / Fetch
1	17:59:14	use empinfo	0 row(s) affected	0.000 sec

36°C Mostly sunny

Search

ENG IN

06:04 PM 18-03-2025

MySQL Workbench

Local instance MySQL80

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

- collage
- empinfo
  - Tables
  - Views
  - Stored Procedures
  - Functions
- kirandb
- librarydb
  - Tables
    - books
    - fines
    - loans
    - reservations
    - users
  - Views
  - Stored Procedures
  - Functions

Administration Schemas

Information

No object selected

Object Info Session

SQL File 1* SQL File 3* SQL File 4* SQL File 5* SQL File 6* SQL File 7* SQL File 8* SQL File 9* LibraryManagementdb

Limit to 1000 rows

6.INSERTING DATA TO USERS

```
47
48  CREATE TABLE Fines (
49      fine_id INT PRIMARY KEY AUTO_INCREMENT,
50      user_id INT,
51      loan_id INT,
52      amount DECIMAL(5,2) NOT NULL,
53      paid BOOLEAN DEFAULT FALSE,
54      FOREIGN KEY (user_id) REFERENCES Users(user_id) ON DELETE CASCADE,
55      FOREIGN KEY (loan_id) REFERENCES Loans(loan_id) ON DELETE CASCADE
56  );
57
58  -- Adding Users
59  INSERT INTO Users (name, email, phone, address, membership_type)
60  VALUES
61  ('Siva', 'siva@example.com', '1234567890', '123 Main St', 'Student'),
62  ('Ram', 'ram@example.com', '9876543210', '456 Elm St', 'Faculty'),
63  ('John', 'john@example.com', '1122334455', '789 Oak St', 'Guest'),
64  ('Priya', 'priya@example.com', '9988776655', '101 Maple St', 'Student'),
65  ('Karan', 'karan@example.com', '7766554433', '202 Birch St', 'Faculty'),
66  ('Meena', 'meena@example.com', '6655443322', '303 Cedar St', 'Guest'),
67  ('Arjun', 'arjun@example.com', '5544332211', '404 Walnut St', 'Student'),
68  ('Ravi', 'ravi@example.com', '4433221100', '505 Spruce St', 'Faculty'),
69  ('Divya', 'divya@example.com', '3322110099', '606 Pine St', 'Guest'),
```

Output

Action Output

#	Time	Action	Message	Duration / Fetch
1	17:59:14	use empinfo	0 row(s) affected	0.000 sec

36°C Mostly sunny

Search

ENG IN

06:05 PM 18-03-2025

MySQL Workbench

Local instance MySQL80

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

- collage
- empinfo
  - Tables
  - Views
  - Stored Procedures
  - Functions
- kirandb
- librarydb
  - Tables
    - books
    - fines
    - loans
    - reservations
    - users
  - Views
  - Stored Procedures
  - Functions

Administration Schemas

Information

No object selected

Object Info Session

SQL File 1* SQL File 3* SQL File 4* SQL File 5* SQL File 6* SQL File 7* SQL File 8* SQL File 9* LibraryManagementdb

Limit to 1000 rows

```
98
99  -- Adding Books
100 INSERT INTO Books (title, author, publisher, published_year, isbn, category, copies_available)
101 VALUES
102 ('The Great Gatsby', 'F. Scott Fitzgerald', 'Scribner', 1925, '9780743273565', 'Fiction', 5),
103 ('To Kill a Mockingbird', 'Harper Lee', 'J.B. Lippincott & Co.', 1960, '9780061120084', 'Fiction', 4),
104 ('1984', 'George Orwell', 'Secker & Warburg', 1949, '9780451524935', 'Dystopian', 6),
105 ('Moby-Dick', 'Herman Melville', 'Harper & Brothers', 1851, '9780142437247', 'Adventure', 3),
106 ('Pride and Prejudice', 'Jane Austen', 'T. Egerton', 1813, '9780679783268', 'Romance', 7),
107 ('The Catcher in the Rye', 'J.D. Salinger', 'Little, Brown and Company', 1951, '9780316769488', 'Fiction', 8),
108 ('The Hobbit', 'J.R.R. Tolkien', 'Allen & Unwin', 1937, '9780345339683', 'Fantasy', 10),
109 ('War and Peace', 'Leo Tolstoy', 'The Russian Messenger', 1869, '9780199232765', 'Historical', 2),
110 ('Crime and Punishment', 'Fyodor Dostoevsky', 'The Russian Messenger', 1866, '9780486415871', 'Philosophical', 5),
111 ('Brave New World', 'Aldous Huxley', 'Chatto & Windus', 1932, '9780060850524', 'Dystopian', 6),
112 ('Les Misérables', 'Victor Hugo', 'A. Lacroix, Verboeckhoven & Cie', 1862, '9780451419439', 'Historical', 5),
113 ('The Brothers Karamazov', 'Fyodor Dostoevsky', 'The Russian Messenger', 1880, '9780374528379', 'Philosophical', 4),
114 ('Wuthering Heights', 'Emily Brontë', 'Thomas Cautley Newby', 1847, '9780486292564', 'Gothic', 6),
115 ('Anna Karenina', 'Leo Tolstoy', 'The Russian Messenger', 1878, '9780143035008', 'Romance', 3),
116 ('Dracula', 'Bram Stoker', 'Archibald Constable', 1897, '9780486454016', 'Horror', 7),
117 ('Frankenstein', 'Mary Shelley', 'Lackington, Hughes, Harding, Mavor & Jones', 1818, '9780486282114', 'Science Fiction', 5),
118 ('One Hundred Years of Solitude', 'Gabriel García Márquez', 'Harper & Row', 1967, '9780060883287', 'Magical Realism', 6),
119 ('Fahrenheit 451', 'Ray Bradbury', 'Ballantine Books', 1953, '9781451673319', 'Dystopian', 5),
120 ('Don Quixote', 'Miguel de Cervantes', 'Francisco de Robles', 1615, '9780060934347', 'Adventure', 3),
```

7.INSERTING DATA TO BOOKS

Output

Action Output

#	Time	Action	Message	Duration / Fetch
1	17:59:14	use empinfo	0 row(s) affected	0.000 sec

36°C Mostly sunny

Search

ENG IN

06:05 PM 18-03-2025

MySQL Workbench

Local instance MySQL80

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

- collage
- empinfo
  - Tables
  - Views
  - Stored Procedures
  - Functions
- kirandb
- librarydb
  - Tables
    - books
    - fines
    - loans
    - reservations
    - users
  - Views
  - Stored Procedures
  - Functions

Administration Schemas

Information

No object selected

Object Info Session

SQL File 1* SQL File 3* SQL File 4* SQL File 5* SQL File 6* SQL File 7* SQL File 8* SQL File 9* LibraryManagementdb

Limit to 1000 rows

8.INSERTING DATA TO LOANS

```
134
135
136     -- borrowing a book
137     INSERT INTO Loans (user_id, book_id, due_date)
138     VALUES
139     (1, 136, DATE_ADD(CURDATE(), INTERVAL 14 DAY)),
140     (2, 137, DATE_ADD(CURDATE(), INTERVAL 14 DAY)),
141     (3, 138, DATE_ADD(CURDATE(), INTERVAL 14 DAY)),
142     (4, 139, DATE_ADD(CURDATE(), INTERVAL 14 DAY)),
143     (5, 140, DATE_ADD(CURDATE(), INTERVAL 14 DAY)),
144     (6, 141, DATE_ADD(CURDATE(), INTERVAL 14 DAY)),
145     (7, 142, DATE_ADD(CURDATE(), INTERVAL 14 DAY)),
146     (8, 143, DATE_ADD(CURDATE(), INTERVAL 14 DAY)),
147     (9, 144, DATE_ADD(CURDATE(), INTERVAL 14 DAY)),
148     (10, 145, DATE_ADD(CURDATE(), INTERVAL 14 DAY)),
149     (11, 146, DATE_ADD(CURDATE(), INTERVAL 14 DAY)),
150     (12, 147, DATE_ADD(CURDATE(), INTERVAL 14 DAY)),
151     (13, 148, DATE_ADD(CURDATE(), INTERVAL 14 DAY)),
152     (14, 149, DATE_ADD(CURDATE(), INTERVAL 14 DAY)),
153     (15, 150, DATE_ADD(CURDATE(), INTERVAL 14 DAY)),
154     (16, 151, DATE_ADD(CURDATE(), INTERVAL 14 DAY)),
155     (17, 152, DATE_ADD(CURDATE(), INTERVAL 14 DAY)),
156     (18, 153, DATE ADD(CURDATE(), INTERVAL 14 DAY)),
```

Output

Action Output

#	Time	Action	Message	Duration / Fetch
1	17:59:14	use empinfo	0 row(s) affected	0.000 sec

36°C Mostly sunny

Search

ENG IN

06:06 PM 18-03-2025

MySQL Workbench

Local instance MySQL80

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

- collage
- empinfo
  - Tables
  - Views
  - Stored Procedures
  - Functions
- kirandb
- librarydb
  - Tables
    - books
    - fines
    - loans
    - reservations
    - users
  - Views
  - Stored Procedures
  - Functions

Administration Schemas

Information

No object selected

Object Info Session

SQL File 1* SQL File 3* SQL File 4* SQL File 5* SQL File 6* SQL File 7* SQL File 8* SQL File 9* LibraryManagementdb

Limit to 1000 rows

```
161  (23, 158, DATE_ADD(CURDATE(), INTERVAL 14 DAY));
162
163
164  -- returning a book with paying fine
165  -- returning a book
166  UPDATE Loans
167  SET return_date = CURDATE(), status = 'Returned'
168  WHERE loan_id = 10;
169
170  -- Update the Fines table to mark the fine as Paid
171  UPDATE Fines
172  SET paid = TRUE
173  WHERE loan_id = 10;
174
175
176
177  -- checking for overdue books
178  SELECT * FROM Loans
179  WHERE due_date < CURDATE() AND status = 'Borrowed';
180
181  -- checking fines
182  SELECT Users.name, Fines.amount, Fines.paid
183  FROM Fines
```

9.UPDATING PAYMENT STATUS

Output

Action Output

#	Time	Action	Message	Duration / Fetch
1	17:59:14	use empinfo	0 row(s) affected	0.000 sec

36°C Mostly sunny

Search

ENG IN

06:06 PM 18-03-2025

MySQL Workbench

Local instance MySQL80

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

- collage
- empinfo
  - Tables
  - Views
  - Stored Procedures
  - Functions
- kirandb
- librarydb
  - Tables
    - books
    - fines
    - loans
    - reservations
    - users
  - Views
  - Stored Procedures
  - Functions

Administration Schemas

Information

No object selected

Object Info Session

SQL File 1* SQL File 3* SQL File 4* SQL File 5* SQL File 6* SQL File 7* SQL File 8* SQL File 9* LibraryManagementdb*

Limit to 1000 rows

10.CHECKING FINES

```
173    WHERE loan_id = 10;
174
175
176
177    -- checking for overdue books
178    SELECT * FROM Loans
179    WHERE due_date < CURDATE() AND status = 'Borrowed';
180
181    -- checking fines
182    SELECT Users.name, Fines.amount, Fines.paid
183    FROM Fines
184    JOIN Users ON Users.user_id = Fines.user_id
185    WHERE Fines.paid = FALSE;
186
187    SELECT Users.name, Fines.amount, Fines.paid
188    FROM Fines
189    JOIN Users ON Users.user_id = Fines.user_id
190    WHERE Fines.paid = TRUE;
191
192    -- Automatically apply fines for overdue books
193    DELIMITER //
194    CREATE TRIGGER after_loan_update
195    AFTER UPDATE ON Loans
```

Output

Action Output

#	Time	Action	Message	Duration / Fetch
1	17:59:14	use empinfo	0 row(s) affected	0.000 sec

36°C Mostly sunny

Search

ENG IN

06:07 PM 18-03-2025

MySQL Workbench

Local instance MySQL80

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

- collage
- empinfo
  - Tables
  - Views
  - Stored Procedures
  - Functions
- kirandb
- librarydb
  - Tables
    - books
    - fines
    - loans
    - reservations
    - users
  - Views
  - Stored Procedures
  - Functions

Administration Schemas

Information

No object selected

Object Info Session

SQL File 1* SQL File 3* SQL File 4* SQL File 5* SQL File 6* SQL File 7* SQL File 8* SQL File 9* LibraryManagementdb*

Limit to 1000 rows

```
192  -- Automatically apply fines for overdue books
193  DELIMITER //
194  CREATE TRIGGER after_loan_update
195  AFTER UPDATE ON Loans
196  FOR EACH ROW
197  BEGIN
198      -- If loan is overdue and still borrowed, insert a fine
199      IF NEW.due_date < CURDATE() AND NEW.status = 'Borrowed' THEN
200          INSERT INTO Fines (user_id, loan_id, amount, paid)
201          VALUES (NEW.user_id, NEW.loan_id, 5.00, FALSE);
202      END IF;
203
204      -- If loan is returned, update fine as paid
205      IF NEW.status = 'Returned' THEN
206          UPDATE Fines
207          SET paid = TRUE, amount = 5.00
208          WHERE loan_id = NEW.loan_id;
209      END IF;
210  END;
211  //
212  DELIMITER ;
213
214  DELIMITER //
```

11.APPLYING FINES FOR OVERDUE BOOKS

Output

Action Output

#	Time	Action	Message	Duration / Fetch
1	17:59:14	use empinfo	0 row(s) affected	0.000 sec

36°C Mostly sunny

Search

ENG IN

06:09 PM 18-03-2025

MySQL Workbench

Local instance MySQL80

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

- collage
- empinfo
  - Tables
  - Views
  - Stored Procedures
  - Functions
- kirandb
- librarydb
  - Tables
    - books
    - fines
    - loans
    - reservations
    - users
  - Views
  - Stored Procedures
  - Functions

Administration Schemas

Information

No object selected

Object Info Session

SQL File 1* SQL File 3* SQL File 4* SQL File 5* SQL File 6* SQL File 7* SQL File 8* SQL File 9* LibraryManagementdb*

Limit to 1000 rows

```
210  END;
211  //
212  DELIMITER ;
213
214  DELIMITER //
215  CREATE TRIGGER after_book_return
216  AFTER UPDATE ON Loans
217  FOR EACH ROW
218  BEGIN
219      IF NEW.status = 'Returned' THEN
220          -- Find the first pending reservation for this book
221          UPDATE Reservations
222          SET status = 'Completed'
223          WHERE book_id = NEW.book_id
224          AND status = 'Pending'
225          ORDER BY reservation_date ASC
226          LIMIT 1;
227      END IF;
228  END;
229  //
230  DELIMITER ;
231
232
```

12.CHECKING STATUS FOR BOOK RETURN

Output

Action Output

#	Time	Action	Message	Duration / Fetch
1	17:59:14	use empinfo	0 row(s) affected	0.000 sec

36°C Mostly sunny

Search

ENG IN

06:09 PM 18-03-2025

MySQL Workbench

Local instance MySQL80

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

- collage
- empinfo
  - Tables
  - Views
  - Stored Procedures
  - Functions
- kirandb
- librarydb
  - Tables
    - books
    - fines
    - loans
    - reservations
    - users
  - Views
  - Stored Procedures
  - Functions

Administration Schemas

Information

No object selected

Object Info Session

SQL File 1* SQL File 3* SQL File 4* SQL File 5* SQL File 6* SQL File 7* SQL File 8* SQL File 9* LibraryManagementdb*

Limit to 1000 rows

```
243
244 •  INSERT INTO Reservations (user_id, book_id)
245    VALUES (3, 136);
246
247 •  SELECT Users.name, Reservations.reservation_date, Reservations.status
248    FROM Reservations
249    JOIN Users ON Reservations.user_id = Users.user_id
250    WHERE Reservations.book_id = 136
251    ORDER BY Reservations.reservation_date ASC;
252
253 •  INSERT INTO Loans (user_id, book_id, due_date)
254    SELECT user_id, book_id, DATE_ADD(CURDATE(), INTERVAL 14 DAY)
255    FROM Reservations
256    WHERE book_id = 136 AND status = 'Completed'
257    ORDER BY reservation_date ASC
258    LIMIT 1;
259
260 •  UPDATE Reservations
261    SET status = 'Cancelled'
262    WHERE reservation_id = 5;
263
264
265
```

13.INSERTING DATA TO RESERVATIONS

Output

Action Output

#	Time	Action	Message	Duration / Fetch
1	17:59:14	use empinfo	0 row(s) affected	0.000 sec

34°C Partly sunny

Search

ENG IN

06:10 PM 18-03-2025

MySQL Workbench

Local instance MySQL80

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

- collage
- empinfo
  - Tables
  - Views
  - Stored Procedures
  - Functions
- kirandb
- **librarydb**
  - Tables
    - books
    - fines
    - loans
    - reservations
    - users
  - Views
  - Stored Procedures
  - Functions

Administration Schemas

Information

No object selected

Object Info Session

SQL File 1* SQL File 3* SQL File 4* SQL File 5* SQL File 6* SQL File 7* SQL File 8* SQL File 9* LibraryManagementdb*

Limit to 1000 rows

```
246
247 • SELECT Users.name, Reservations.reservation_date, Reservations.status
248   FROM Reservations
249   JOIN Users ON Reservations.user_id = Users.user_id
250   WHERE Reservations.book_id = 136
251   ORDER BY Reservations.reservation_date ASC;
252
253 • INSERT INTO Loans (user_id, book_id, due_date)
254   SELECT user_id, book_id, DATE_ADD(CURDATE(), INTERVAL 14 DAY)
255   FROM Reservations
256   WHERE book_id = 136 AND status = 'Completed'
257   ORDER BY reservation_date ASC
258   LIMIT 1;
259
```

14.GETTING RESERVATIONS DATA

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

name	reservation_date	status
John	2025-03-18 16:17:14	Cancelled

Result Grid Form Editor

Result 3

Read Only

Output

Action Output

#	Time	Action	Message	Duration / Fetch
5	18:12:15	UPDATE Reservations SET status = 'Cancelled' WHERE reservation_id = 1	1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0	0.016 sec
6	18:12:27	select * from reservations LIMIT 0, 1000	1 row(s) returned	0.000 sec / 0.000 sec
7	18:13:44	SELECT Users.name, Reservations.reservation_date, Reservations.status FROM Reservations JOIN Users O...	1 row(s) returned	0.000 sec / 0.000 sec

34°C Partly sunny

Search

ENG IN

06:14 PM 18-03-2025

MySQL Workbench

Local instance MySQL80

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

- collage
- empinfo
  - Tables
  - Views
  - Stored Procedures
  - Functions
- kirandb
- **librarydb**
  - Tables
    - books
    - fines
    - loans
    - reservations
    - users
  - Views
  - Stored Procedures
  - Functions

Administration Schemas

Information

No object selected

Object Info Session

SQL File 1* SQL File 3* SQL File 4* SQL File 5* SQL File 6* SQL File 7* SQL File 8* SQL File 9* LibraryManagementdb*

Limit to 1000 rows

```
244 • INSERT INTO Reservations (user_id, book_id)
245   VALUES (3, 136);
246
247 • SELECT Users.name, Reservations.reservation_date, Reservations.status
248   FROM Reservations
249   JOIN Users ON Reservations.user_id = Users.user_id
250   WHERE Reservations.book_id = 136
251   ORDER BY Reservations.reservation_date ASC;
252
253 • INSERT INTO Loans (user_id, book_id, due_date)
254   SELECT user_id, book_id, DATE_ADD(CURDATE(), INTERVAL 14 DAY)
255   FROM Reservations
256   WHERE book_id = 136 AND status = 'Completed'
257   ORDER BY reservation_date ASC
258   LIMIT 1;
259
260 • UPDATE Reservations
261   SET status = 'Cancelled'
262   WHERE reservation_id = 1;
263
264
265
```

15.INSERTING DATA TO LOANS BASED ON RESERVATION

Output

Action Output

#	Time	Action	Message	Duration / Fetch
6	18:12:27	select * from reservations LIMIT 0, 1000	1 row(s) returned	0.000 sec / 0.000 sec
7	18:13:44	SELECT Users.name, Reservations.reservation_date, Reservations.status FROM Reservations JOIN Users O...	1 row(s) returned	0.000 sec / 0.000 sec
8	18:14:46	INSERT INTO Loans (user_id, book_id, due_date) SELECT user_id, book_id, DATE_ADD(CURDATE(), INTE...	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0	0.000 sec

34°C Partly sunny

Search

ENG IN

06:14 PM 18-03-2025

MySQL Workbench

Local instance MySQL80

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

- collage
- empinfo
  - Tables
  - Views
  - Stored Procedures
  - Functions
- kirandb
- **librarydb**
  - Tables
    - books
    - fines
    - loans
    - reservations
    - users
  - Views
  - Stored Procedures
  - Functions

Administration Schemas

Information

No object selected

Object Info Session

SQL File 1* SQL File 3* SQL File 4* SQL File 5* SQL File 6* SQL File 7* SQL File 8* SQL File 9* LibraryManagementdb*

Limit to 1000 rows

```
244 • INSERT INTO Reservations (user_id, book_id)
245   VALUES (3, 136);
246
247 • SELECT Users.name, Reservations.reservation_date, Reservations.status
248   FROM Reservations
249   JOIN Users ON Reservations.user_id = Users.user_id
250   WHERE Reservations.book_id = 136
251   ORDER BY Reservations.reservation_date ASC;
252
253 • INSERT INTO Loans (user_id, book_id, due_date)
254   SELECT user_id, book_id, DATE_ADD(CURDATE(), INTERVAL 14 DAY)
255   FROM Reservations
256   WHERE book_id = 136 AND status = 'Completed'
257   ORDER BY reservation_date ASC
258   LIMIT 1;
259
260 • UPDATE Reservations
261   SET status = 'Cancelled'
262   WHERE reservation_id = 1;
263
264
265
```

16.UPDATING RESERVATIONS STATUS

Output

Action Output

#	Time	Action	Message	Duration / Fetch
6	18:12:27	select * from reservations LIMIT 0, 1000	1 row(s) returned	0.000 sec / 0.000 sec
7	18:13:44	SELECT Users.name, Reservations.reservation_date, Reservations.status FROM Reservations JOIN Users O...	1 row(s) returned	0.000 sec / 0.000 sec
8	18:14:46	INSERT INTO Loans (user_id, book_id, due_date) SELECT user_id, book_id, DATE_ADD(CURDATE(), INTE...	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0	0.000 sec

34°C Partly sunny

Search

ENG IN

06:15 PM 18-03-2025