

## EXTRA LAB PRACTISE FOR DATABASE CONCEPTS

### 1. Introduction to SQL

**Lab 3 :** Create a database called library\_db and a table books with columns: book\_id, title, author, publisher, year\_of\_publication, and price. Insert five records into the table.

- CREATE DATABASE library\_db
- CREATE TABLE books (book\_id int PRIMARY KEY AUTO\_INCREMENT, title varchar(20),author varchar(20),publisher varchar(20),year\_of\_publication varchar(20), price int)
- INSERT INTO books (title,author,publisher,year\_of\_publication,price)VALUES('The Alchemist', 'Paulo Coelho', 'HarperCollins', 1988, 299.50),('Clean Code', 'Robert C. Martin', 'Pearson', 2008, 550.00),('Atomic Habits', 'James Clear', 'Penguin', 2018, 450.75),('Wings of Fire', 'A. P. J. Abdul Kalam', 'Universities Press', 1999, 350.00),('Think and Grow Rich', 'Napoleon Hill', 'Rupa', 1937, 399.99);

The screenshot shows the phpMyAdmin interface with the following details:

- Server:** 127.0.0.1
- Database:** library\_db
- Table:** books
- Actions:** Browse, Structure, SQL, Search, Insert, Export, Import, Privileges, Operations
- Message:** Showing rows 0 - 4 (5 total, Query took 0.0003 seconds.)
- SQL Query:** SELECT \* FROM `books`
- Table Headers:** book\_id, title, author, publisher, year\_of\_publication, price
- Data Rows:**

book_id	title	author	publisher	year_of_publication	price
1	The Alchemist	Paulo Coelho	HarperCollins	1988	300
2	Clean Code	Robert C. Martin	Pearson	2008	550
3	Atomic Habits	James Clear	Penguin	2018	450.75
4	Wings of Fire	A. P. J. Abdul Kalam	Universities Press	1999	350
5	Think and Grow Rich	Napoleon Hill	Rupa	1937	399.99

**Lab 4 :** Create a table members in library\_db with columns: member\_id, member\_name, date\_of\_membership, and email. Insert five records into this table.

- CREATE TABLE members (member\_id INT PRIMARY KEY, member\_name VARCHAR(20), date\_of\_membership DATE, email VARCHAR(20));
- INSERT INTO members (member\_name, date\_of\_membership, email) VALUES ('Ayush Patel', '2024-01-15', 'ayush@gmail.com'), ('Meet Shah', '2020-02-10', 'meet@gmail.com'), ('Karan Mehta', '2024-03-05', 'karan@gmail.com'), ('Dev Joshi', '2021-04-12', 'dev@gmail.com'), ('Vishal Kumar', '2020-05-20', 'vishal@gmail.com');

	member_id	member_name	date_of_membership	email
<input type="checkbox"/>	1	Ayush Patel	2024-01-15	ayush@gmail.com
<input type="checkbox"/>	2	Meet Shah	2020-02-10	meet@gmail.com
<input type="checkbox"/>	3	Karan Mehta	2024-03-05	karan@gmail.com
<input type="checkbox"/>	4	Dev Joshi	2021-04-12	dev@gmail.com
<input type="checkbox"/>	5	Vishal Kumar	2020-05-20	vishal@gmail.com

## 2. SQL Syntax

**Lab 3 :** Retrieve all members who joined the library before 2022. Use appropriate SQL syntax with WHERE and ORDER BY.

- SELECT \* FROM `members` WHERE date\_of\_membership < '2022-01-01' ORDER BY date\_of\_membership

	member_id	member_name	date_of_membership	email
<input type="checkbox"/>	2	Meet Shah	2020-02-10	meet@gmail.com
<input type="checkbox"/>	5	Vishal Kumar	2020-05-20	vishal@gmail.com
<input type="checkbox"/>	4	Dev Joshi	2021-04-12	dev@gmail.com

**Lab 4 :** Write SQL queries to display the titles of books published by a specific author. Sort the results by year\_of\_publication in descending order.

- SELECT title FROM books WHERE author = 'Paulo Coelho' ORDER BY year\_of\_publication DESC

The screenshot shows the MySQL Workbench interface with the following details:

- Server: 127.0.0.1
- Database: library\_db
- Table: books
- Query results:

```
SELECT title FROM books WHERE author = 'Paulo Coelho' ORDER BY year_of_publication DESC;
```
- Message bar: Showing rows 0 - 0 (1 total, Query took 0.0004 seconds.)
- Table view:

title
The Alchemist

### 3. SQL Constraints

**Lab 3 :** Add a CHECK constraint to ensure that the price of books in the books table is greater than 0.

- ALTER TABLE books ADD CONSTRAINT check\_price\_positive CHECK (price > 0)

The screenshot shows the MySQL Workbench interface with the following details:

- Server: 127.0.0.1
- Database: library\_db
- Table: books
- Data entry screen for a new book record:

publisher	varchar(20)	Rupa
year_of_publication	varchar(20)	1937
price	int(11)	-200

- An error message box is displayed:

Error

SQL query: [Copy](#) [Edit](#)

```
UPDATE `books` SET `price` = '-200' WHERE `books`
```

MySQL said: [?](#)

#4025 - CONSTRAINT `chk\_price\_positive` failed for `library\_db`.`books`

- Below the error message, the history shows the failed update command:

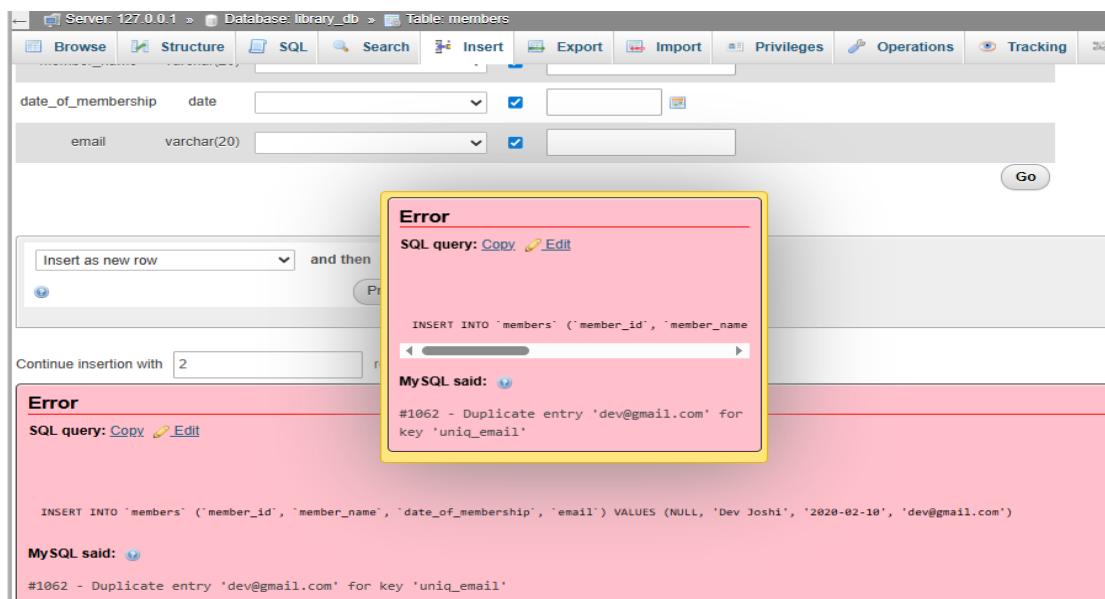
```
UPDATE `books` SET `price` = '-200' WHERE `books`.`book_id` = 5
```

MySQL said: [?](#)

#4025 - CONSTRAINT `chk\_price\_positive` failed for `library\_db`.`books`

**Lab 4 :** Modify the members table to add a UNIQUE constraint on the email column, ensuring that each member has a unique email address.

- ALTER TABLE members ADD CONSTRAINT uniq\_email UNIQUE(email)



## 4. Main SQL Commands and Sub-commands (DDL)

**Lab 3 :** Create a table authors with the following columns: author\_id, first\_name, last\_name, and country. Set author\_id as the primary key.

- CREATE TABLE authors (author\_id int PRIMARY KEY AUTO\_INCREMENT, last\_name varchar(50),country varchar(20))

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	author_id	int(11)			No	None		AUTO_INCREMENT	More
2	last_name	varchar(50)	utf8mb4_general_ci		Yes	NULL			More
3	country	varchar(20)	utf8mb4_general_ci		Yes	NULL			More

Check all With selected: Primary Unique Index Spatial Fulltex

1 column(s) after country Go

**Lab 4 :** Create a table publishers with columns: publisher\_id, publisher\_name, contact\_number, and address. Set publisher\_id as the primary key and contact\_number as unique.

- CREATE TABLE publishers (publisher\_id int PRIMARY KEY AUTO\_INCREMENT,publisher\_name varchar(50),contact\_number int UNIQUE KEY, address varchar(50))

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	publisher_id	int(11)			No	None		AUTO_INCREMENT	Change  Drop  More
2	publisher_name	varchar(50)	utf8mb4_general_ci		Yes	NULL			Change  Drop  More
3	contact_number	int(11)			Yes	NULL			Change  Drop  More
4	address	varchar(50)	utf8mb4_general_ci		Yes	NULL			Change  Drop  More

## 5. ALTER Command

**Lab 3 :** Add a new column genre to the books table. Update the genre for all existing records.

- ALTER TABLE books ADD COLUMN genre varchar(50)
- UPDATE books SET genre = 'Fiction / Philosophy' WHERE book\_id = 1;
- UPDATE books SET genre = 'Programming' WHERE book\_id = 2;
- UPDATE books SET genre = 'Self Help' WHERE book\_id = 3;
- UPDATE books SET genre = 'Biography' WHERE book\_id = 4;
- UPDATE books SET genre = 'Self Help' WHERE book\_id = 5;

book_id	title	author	publisher	year_of_publication	price	genre
1	The Alchemist	Paulo Coelho	HarperCollins	1988	300	Fiction / Philosophy
2	Clean Code	Robert C. Martin	Pearson	2008	550	Programming
3	Atomic Habits	James Clear	Penguin	2018	451	Self Help
4	Wings of Fire	A. P. J. Abdul Kalam	Universities Press	1999	350	Biography
5	Think and Grow Rich	Napoleon Hill	Rupa	1937	400	Self Help

**Lab 4 :** Modify the members table to increase the length of the email column to 100 characters.

- ALTER TABLE members MODIFY email varchar(100)

The screenshot shows the MySQL Workbench interface with the following details:

- Server: 127.0.0.1
- Database: library\_db
- Table: members
- Table structure view is selected.
- Columns listed:

  - member\_id (int(11), primary key, auto-increment)
  - member\_name (varchar(20))
  - date\_of\_membership (date)
  - email (varchar(100))

- Action column contains edit, drop, and more options for each row.

## 6. DROP Command

**Lab 3 :** Drop the publishers table from the database after verifying its structure.

- DROP TABLE publishers

The screenshot shows the MySQL Workbench interface with the following details:

- SQL query: `SELECT * FROM `publishers` WHERE 1 LIMIT 0, 25`
- MySQL said: #1146 - Table 'library\_db.publishers' doesn't exist
- Error message: #1146 - Table 'library\_db.publishers' doesn't exist

**Lab 4** : Create a backup of the members table and then drop the original members table.

- CREATE TABLE members\_backup AS SELECT \* FROM members
- DROP TABLE members

The screenshot shows the phpMyAdmin interface for a database named 'library\_db'. The current table is 'members\_backup'. The top navigation bar includes 'Server: 127.0.0.1', 'Database: library\_db', and 'Table: members\_backup'. Below the navigation are tabs for 'Browse', 'Structure', 'SQL', 'Search', 'Insert', 'Export', 'Import', and 'Privileges'. A warning message states: 'Current selection does not contain a unique column. Grid edit, checkbox, Edit, Copy and Delete features are not available'. A success message below it says: 'Showing rows 0 - 4 (5 total, Query took 0.0005 seconds.)'. The SQL query shown is 'SELECT \* FROM `members\_backup`'. There are buttons for 'Profiling', 'Edit inline', 'Edit', 'Explain SQL', 'Create PHP code', and 'Refresh'. Below these are filters for 'Show all' (unchecked), 'Number of rows: 25', and a search bar 'Filter rows: Search this table'. An 'Extra options' button is also present. The main content area displays a table with the following data:

member_id	member_name	date_of_membership	email
1	Ayush Patel	2024-01-15	ayush@gmail.com
2	Meet Shah	2020-02-10	meet@gmail.com
3	Karan Mehta	2024-03-05	karan@gmail.com
4	Dev Joshi	2021-04-12	dev@gmail.com
5	Vishal Kumar	2020-05-20	vishal@gmail.com

## 7. Data Manipulation Language (DML)

**Lab 4 :** Insert three new authors into the authors table, then update the last name of one of the authors.

- INSERT INTO authors (first\_name,last\_name,country) VALUES ('Paulo', 'Coelho', 'Brazil');
- INSERT INTO authors (first\_name,last\_name,country) VALUES ('Robert', 'Martin', 'USA');
- INSERT INTO authors (first\_name,last\_name,country) VALUES ('James', 'Clear', 'USA');
- UPDATE authors SET last\_name = 'C.martin' WHERE author\_id = 2

The image shows two screenshots of the phpMyAdmin interface, illustrating the state of the 'authors' table in the 'library\_db' database.

**Screenshot 1 (Top):** This screenshot shows the 'authors' table before any updates were made. The table has columns: author\_id, first\_name, last\_name, and country. There are three rows:

author_id	first_name	last_name	country
1	Paulo	Coelho	Brazil
2	Robert	Martin	USA
3	James	Clear	USA

**Screenshot 2 (Bottom):** This screenshot shows the 'authors' table after the update. The row where author\_id = 2 has been updated. The 'last\_name' column now contains 'C.martin'.

author_id	first_name	last_name	country
1	Paulo	Coelho	Brazil
2	Robert	C.martin	USA
3	James	Clear	USA

**Lab 5 :** Delete a book from the books table where the price is higher than \$100.

- DELETE FROM books WHERE price>100

The screenshot shows the phpMyAdmin interface for the 'books' table in the 'library\_db' database. The table has columns: book\_id, title, author, publisher, year\_of\_publication, price, and genre. The data is as follows:

book_id	title	author	publisher	year_of_publication	price	genre
1	The Alchemist	Paulo Coelho	HarperCollins	1988	60	Fiction / Philosophy
3	Atomic Habits	James Clear	Penguin	2018	95	Self Help
5	Think and Grow Rich	Napoleon Hill	Rupa	1937	90	Self Help

## 8. UPDATE Command

**Lab 3 :** Update the year\_of\_publication of a book with a specific book\_id.

- UPDATE books SET year\_of\_publication = 2020 WHERE book\_id = 3

The screenshot shows the phpMyAdmin interface for the 'books' table in the 'library\_db' database after the update query was run. The 'year\_of\_publication' for book\_id 3 has been updated to 2020. The data is as follows:

book_id	title	author	publisher	year_of_publication	price	genre
1	The Alchemist	Paulo Coelho	HarperCollins	1988	60	Fiction / Philosophy
3	Atomic Habits	James Clear	Penguin	2020	95	Self Help
5	Think and Grow Rich	Napoleon Hill	Rupa	1937	90	Self Help

#### **Lab 4 : Increase the price of all books published before 2015 by 10%.**

- UPDATE books SET price = price \* 1.10 WHERE year\_of\_publication < 2015

The screenshot shows the MySQL Workbench interface with the following details:

- Server:** 127.0.0.1
- Database:** library\_db
- Table:** books
- Query Result:** Showing rows 0 - 2 (3 total, Query took 0.0003 seconds.)
- SQL Query:** SELECT \* FROM `books`
- Table Data:**

	book_id	title	author	publisher	year_of_publication	price	genre
<input type="checkbox"/>	1	The Alchemist	Paulo Coelho	HarperCollins	1988	66	Fiction / Philosophy
<input type="checkbox"/>	3	Atomic Habits	James Clear	Penguin	2020	95	Self Help
<input type="checkbox"/>	5	Think and Grow Rich	Napoleon Hill	Rupa	1937	99	Self Help

#### **9. DELETE Command**

#### **Lab 3 : Remove all members who joined before 2020 from the members table.**

- DELETE FROM members WHERE date\_of\_membership < '2020-01-01'

The screenshot shows the MySQL Workbench interface with the following details:

- Server:** 127.0.0.1
- Database:** library\_db
- Table:** members
- Query Result:** Showing rows 0 - 4 (5 total, Query took 0.0005 seconds.)
- SQL Query:** SELECT \* FROM `members`
- Table Data:**

	member_id	member_name	date_of_membership	email
<input type="checkbox"/>	1	Ayush Patel	2024-01-15	ayush@gmail.com
<input type="checkbox"/>	2	Meet Shah	2010-02-10	meet@gmail.com
<input type="checkbox"/>	3	Karan Mehta	2005-03-05	karan@gmail.com
<input type="checkbox"/>	4	Dev Joshi	2021-04-12	dev@gmail.com
<input type="checkbox"/>	5	Vishal Kumar	2020-05-20	vishal@gmail.com
- Query Result:** Showing rows 0 - 2 (3 total, Query took 0.0003 seconds.)
- SQL Query:** SELECT \* FROM `members`
- Table Data:**

	member_id	member_name	date_of_membership	email
<input type="checkbox"/>	1	Ayush Patel	2024-01-15	ayush@gmail.com
<input type="checkbox"/>	4	Dev Joshi	2021-04-12	dev@gmail.com
<input type="checkbox"/>	5	Vishal Kumar	2020-05-20	vishal@gmail.com

**Lab 4 :** Delete all books that have a NULL value in the author column.

- DELETE FROM books WHERE author IS null

Showing rows 0 - 1 (2 total, Query took 0.0005 seconds.)

SELECT \* FROM `books`

Profiling [ Edit inline ] [ Edit ] [ Explain SQL ] [ Create PHP code ] [ Refresh ]

Show all Number of rows: 25 Filter rows: Search this table Sort by key: None

	book_id	title	author	publisher	year_of_publication	price	genre
<input type="checkbox"/>	3	Atomic Habits	NULL	Penguin	2020	95	Self Help
<input type="checkbox"/>	5	Think and Grow Rich	Napoleon Hill	Rupa	1937	99	Self Help

Showing rows 0 - 0 (1 total, Query took 0.0003 seconds.)

SELECT \* FROM `books`

Profiling [ Edit inline ] [ Edit ] [ Explain SQL ] [ Create PHP code ] [ Refresh ]

Show all Number of rows: 25 Filter rows: Search this table

	book_id	title	author	publisher	year_of_publication	price	genre
<input type="checkbox"/>	5	Think and Grow Rich	Napoleon Hill	Rupa	1937	99	Self Help

## 10. Data Query Language (DQL)

**Lab 4 :** Write a query to retrieve all books with price between \$50 and \$100.

- SELECT \* FROM books WHERE price BETWEEN 50 and 100

The screenshot shows two separate queries in the SQL tab of phpMyAdmin:

**Query 1 (Top):**

```
SELECT * FROM `books`
```

**Query 2 (Bottom):**

```
SELECT * FROM books WHERE price BETWEEN 50 and 100;
```

Both queries return results for the following books:

book_id	title	author	publisher	year_of_publication	price	genre
1	The Alchemist	Paulo Coelho	HarperCollins	1988	60	Fiction / Philosophy
2	Clean Code	Robert C. Martin	Pearson	2008	550	Programming
3	Atomic Habits	James Clear	Penguin	2018	90	Self Help
4	Wings of Fire	A. P. J. Abdul Kalam	Universities Press	1999	95	Biography
5	Think and Grow Rich	Napoleon Hill	Rupa	1937	400	Self Help

**Lab 5 :** Retrieve the list of books sorted by author in ascending order and limit the results to the top 3 entries.

- SELECT \* FROM `books` ORDER BY author ASC LIMIT 3;

The screenshot shows a single query in the SQL tab of phpMyAdmin:

```
SELECT * FROM `books` ORDER BY author ASC LIMIT 3;
```

The results show the top 3 books by author:

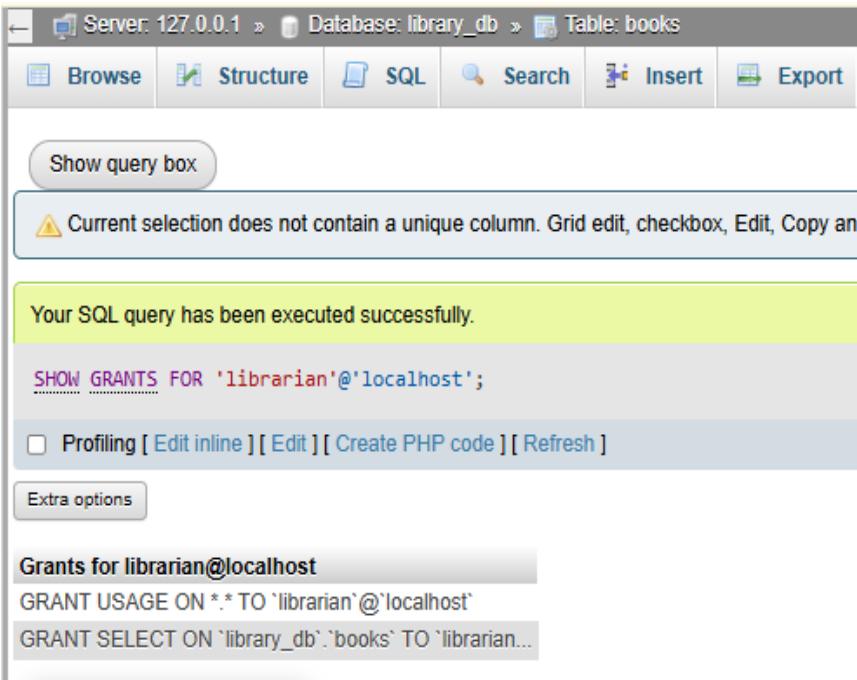
book_id	title	author	publisher	year_of_publication	price	genre
4	Wings of Fire	A. P. J. Abdul Kalam	Universities Press	1999	95	Biography
3	Atomic Habits	James Clear	Penguin	2018	90	Self Help
5	Think and Grow Rich	Napoleon Hill	Rupa	1937	400	Self Help

## 11. Data Control Language (DCL)

**Lab 3 :** Grant SELECT permission to a user named librarian on the books table.

**Lab 4 :** Grant INSERT and UPDATE permissions to the user admin on the members table.

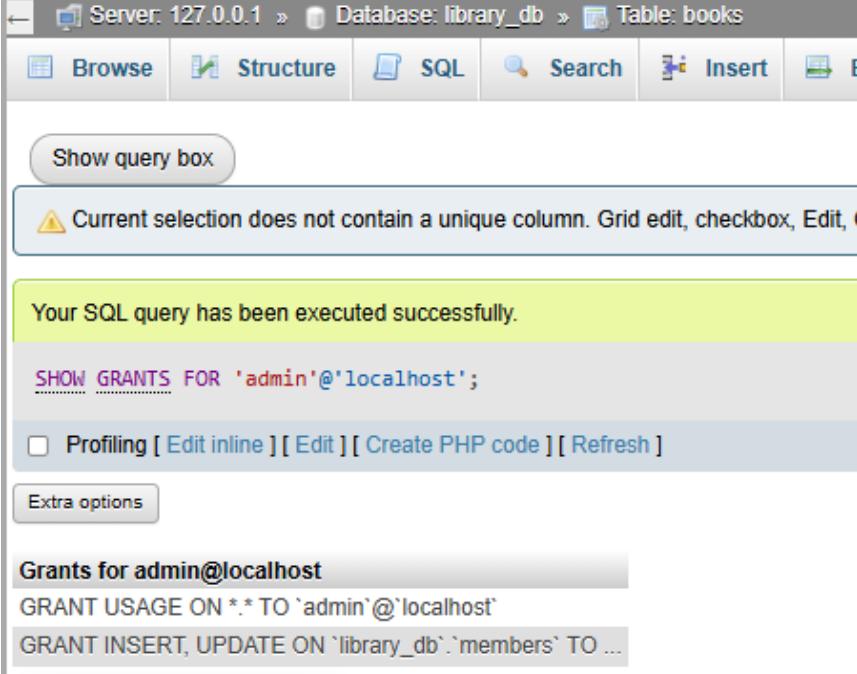
- GRANT SELECT ON library\_db.books TO 'librarian'@'localhost';
- GRANT INSERT, UPDATE ON library\_db.members TO 'admin'@'localhost';



The screenshot shows the phpMyAdmin interface for the 'books' table in the 'library\_db' database. The SQL tab contains the command: `SHOW GRANTS FOR 'librarian'@'localhost';`. The results pane displays:

```
GRANT USAGE ON *.* TO 'librarian'@'localhost'  
GRANT SELECT ON `library_db`.'books' TO 'librarian'@'localhost'
```

Below the results, it says "Your SQL query has been executed successfully."

The screenshot shows the phpMyAdmin interface for the 'books' table in the 'library\_db' database. The SQL tab contains the command: `SHOW GRANTS FOR 'admin'@'localhost';`. The results pane displays:

```
GRANT USAGE ON *.* TO 'admin'@'localhost'  
GRANT INSERT, UPDATE ON `library_db`.'members' TO 'admin'@'localhost'
```

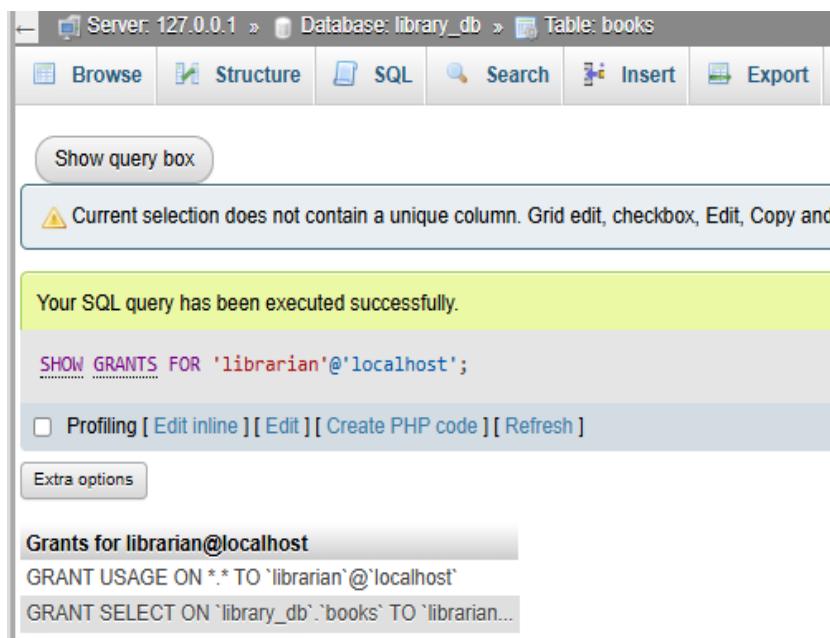
Below the results, it says "Your SQL query has been executed successfully."

## 12. REVOKE Command

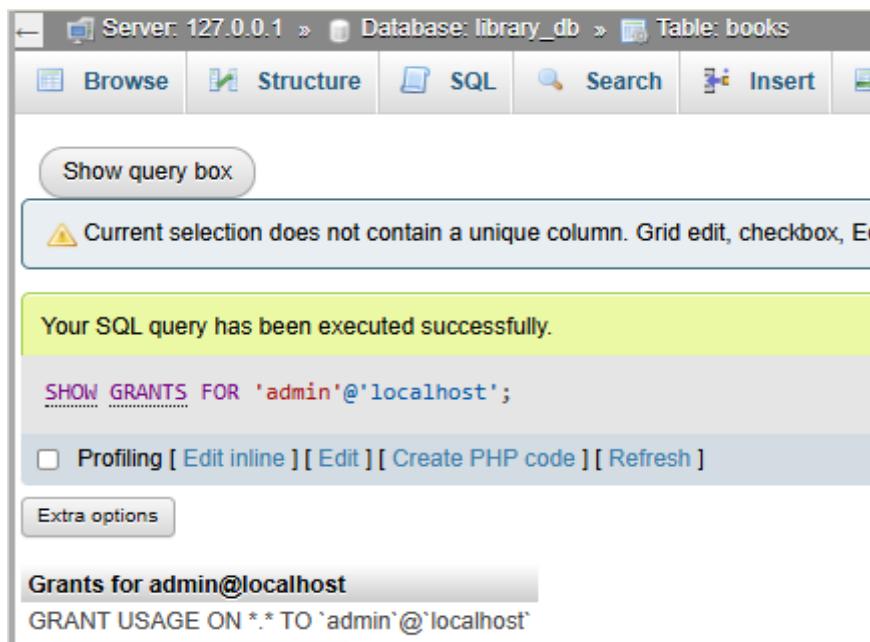
**Lab 3 :** Revoke the INSERT privilege from the user librarian on the books table.

**Lab 4 :** Revoke all permissions from user admin on the members table.

- REVOKE INSERT ON library\_db.books FROM 'librarian'@'localhost';
- REVOKE ALL PRIVILEGES ON library\_db.members FROM 'admin'@'localhost';



The screenshot shows the phpMyAdmin interface for the 'books' table in the 'library\_db' database. The top navigation bar shows 'Server: 127.0.0.1', 'Database: library\_db', and 'Table: books'. Below the navigation bar is a toolbar with 'Browse', 'Structure', 'SQL', 'Search', 'Insert', and 'Export' buttons. A 'Show query box' button is also present. A warning message in a yellow box states: 'Current selection does not contain a unique column. Grid edit, checkbox, Edit, Copy and Paste operations are disabled.' A green success message box below it says: 'Your SQL query has been executed successfully.' The SQL query shown is: `SHOW GRANTS FOR 'librarian'@'localhost';`. Below the query is a 'Profiling' section with options: 'Edit inline', 'Edit', 'Create PHP code', and 'Refresh'. An 'Extra options' button is also visible. The grants for 'librarian'@'localhost' are listed under 'Grants for librarian@localhost':  
GRANT USAGE ON \*.\* TO 'librarian'@'localhost'  
GRANT SELECT ON `library\_db`.`books` TO 'librarian'@'localhost'



The screenshot shows the phpMyAdmin interface for the 'books' table in the 'library\_db' database. The top navigation bar shows 'Server: 127.0.0.1', 'Database: library\_db', and 'Table: books'. Below the navigation bar is a toolbar with 'Browse', 'Structure', 'SQL', 'Search', 'Insert', and 'Export' buttons. A 'Show query box' button is also present. A warning message in a yellow box states: 'Current selection does not contain a unique column. Grid edit, checkbox, Edit, Copy and Paste operations are disabled.' A green success message box below it says: 'Your SQL query has been executed successfully.' The SQL query shown is: `SHOW GRANTS FOR 'admin'@'localhost';`. Below the query is a 'Profiling' section with options: 'Edit inline', 'Edit', 'Create PHP code', and 'Refresh'. An 'Extra options' button is also visible. The grants for 'admin'@'localhost' are listed under 'Grants for admin@localhost':  
GRANT USAGE ON \*.\* TO 'admin'@'localhost'

### 13. Transaction Control Language (TCL)

**Lab 3 :** Use COMMIT after inserting multiple records into the books table, then make another insertion and perform a ROLLBACK.

```
- START TRANSACTION;  
  INSERT INTO books  
  (book_id, title, author, publisher, year_of_publication, price, genre)  
  VALUES  
  (6, 'Data Structures', 'Mark Allen Weiss', 'Pearson', 2014, 450,  
  'Programming');  
  INSERT INTO books  
  (book_id, title, author, publisher, year_of_publication, price, genre)  
  VALUES  
  (7, 'Operating System Concepts', 'Silberschatz', 'Wiley', 2018, 600,  
  'Programming');  
  COMMIT;  
  START TRANSACTION;  
  INSERT INTO books  
  (book_id, title, author, publisher, year_of_publication, price, genre)  
  VALUES  
  (8, 'Machine Learning', 'Tom Mitchell', 'McGraw Hill', 2017, 700, 'AI');  
  ROLLBACK;
```

The screenshot shows the MySQL Workbench interface with the following details:

- Server:** 127.0.0.1
- Database:** library\_db
- Table:** books
- Toolbar:** Browse, Structure, SQL, Search, Insert, Export, Import, Privileges, Operations, Tracking, Trig
- Status Bar:** Showing rows 0 - 6 (7 total, Query took 0.0004 seconds.)
- SQL Editor:** SELECT \* FROM `books`
- Buttons:** Profiling [ Edit inline ] [ Edit ] [ Explain SQL ] [ Create PHP code ] [ Refresh ]
- Table Filter:** Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None
- Table Data:** A grid showing 7 rows of book information. The columns are: book\_id, title, author, publisher, year\_of\_publication, price, and genre. The data includes books like 'The Alchemist' by Paulo Coelho, 'Clean Code' by Robert C. Martin, and 'Data Structures' by Mark Allen Weiss.

**Lab 4 :** Set a SAVEPOINT before making updates to the members table, perform some updates, and then roll back to the SAVEPOINT.

- START TRANSACTION;
- SAVEPOINT before\_update;
- UPDATE members
- SET email = 'updated\_ayush@gmail.com'
- WHERE member\_id = 1;
- UPDATE members
- SET member\_name = 'Dev Joshi Patel'
- WHERE member\_id = 4;
- ROLLBACK TO before\_update;
- COMMIT;

The screenshot shows the phpMyAdmin interface for the 'members' table. The table has columns: member\_id, member\_name, date\_of\_membership, and email. The data is as follows:

	member_id	member_name	date_of_membership	email
<input type="checkbox"/>	1	Ayush Patel	2024-01-15	ayush@gmail.com
<input type="checkbox"/>	4	Dev Joshi	2021-04-12	dev@gmail.com
<input type="checkbox"/>	5	Vishal Kumar	2019-05-20	vishal@gmail.com
<input type="checkbox"/>	6	Jeel Kumar	2015-02-10	jeel@gmail.com

## 14. SQL Joins

**Lab 3 :** Perform an INNER JOIN between books and authors tables to display the title of books and their respective authors' names.

- `SELECT title, first_name, last_name FROM books JOIN authors ON author = CONCAT(first_name, ' ', last_name);`

The screenshot shows the phpMyAdmin interface for a MySQL database named 'library\_db'. The current table is 'books'. The query results show three rows of data from the 'books' table:

title	first_name	last_name
The Alchemist	Paulo	Coelho
Atomic Habits	James	Clear
Think and Grow Rich	Napoleon	Hill

**Lab 4 :** Use a FULL OUTER JOIN to retrieve all records from the books and authors tables, including those with no matching entries in the other table.

- `SELECT title, first_name, last_name FROM books LEFT JOIN authors ON author = CONCAT(first_name, ' ', last_name) UNION SELECT title, first_name, last_name FROM books RIGHT JOIN authors ON author = CONCAT(first_name, ' ', last_name);`

The screenshot shows the phpMyAdmin interface for a MySQL database named 'library\_db'. The current table is 'books'. The query results show six rows of data, including entries from both the 'books' and 'authors' tables:

title	first_name	last_name
The Alchemist	Paulo	Coelho
Atomic Habits	James	Clear
Think and Grow Rich	Napoleon	Hill
Clean Code	NULL	NULL
Wings of Fire	NULL	NULL
NULL	Robert	Martin

## 15. SQL Group By

**Lab 3 :** Group books by genre and display the total number of books in each genre.

- SELECT genre, COUNT(\*) as total\_book FROM books GROUP BY genre;

The screenshot shows the phpMyAdmin interface with the following details:

- Server:** 127.0.0.1
- Database:** library\_db
- Table:** books
- Toolbar:** Browse, Structure, SQL, Search, Insert, Export, Import, Privileges
- Query Result:** Shows the output of the SQL query: "SELECT genre, COUNT(\*) as total\_book FROM books GROUP BY genre;". The result is:

genre	total_book
Biography	1
Fiction / Philosophy	1
Programming	1
Self Help	2
- Buttons:** Profiling [ Edit inline ] [ Edit ] [ Explain SQL ] [ Create PHP code ] [ Refresh ]
- Table Options:** Show all, Number of rows: 25, Filter rows: Search this table
- Extra Options:** Extra options button

**Lab 4 :** Group members by the year they joined and find the number of members who joined each year.

- SELECT date\_of\_membership AS join\_year, COUNT(\*) AS total\_member FROM members GROUP BY date\_of\_membership ORDER BY join\_year;

The screenshot shows the phpMyAdmin interface with the following details:

- Server:** 127.0.0.1
- Database:** library\_db
- Table:** members
- Toolbar:** Browse, Structure, SQL, Search, Insert, Export, Import, Privileges, Operations, Tracking
- Query Result:** Shows the output of the SQL query: "SELECT date\_of\_membership AS join\_year, COUNT(\*) AS total\_member FROM members GROUP BY date\_of\_membership ORDER BY join\_year;". The result is:

join_year	total_member
2020-05-20	1
2021-04-12	1
2024-01-15	1
- Buttons:** Profiling [ Edit inline ] [ Edit ] [ Explain SQL ] [ Create PHP code ] [ Refresh ]
- Table Options:** Show all, Number of rows: 25, Filter rows: Search this table
- Extra Options:** Extra options button

## 16. SQL Stored Procedure

**Lab 3 :** Write a stored procedure to retrieve all books by a particular author.

```
- DELIMITER //
CREATE PROCEDURE getbook_by_author (IN author_name varchar(50))
BEGIN
SELECT * FROM books WHERE author = author_name;
END //
DELIMITER ;
```

The screenshot shows the phpMyAdmin interface for a database named 'library\_db'. The 'Routines' tab is selected. A success message in a green box states: 'Your SQL query has been executed successfully. 1 row affected by the last statement inside the procedure.' Below this, the SQL query `SET @p0='James clear'; CALL `getbook\_by\_author` (@p0);` is shown. The results of the execution are displayed in a table:

book_id	title	author	publisher	year_of_publication	price	genre
3	Atomic Habits	James Clear	Penguin	2018	90	Self Help

### Routines

<input type="checkbox"/> Check all	Export	Drop
Name	Type	Returns
getbook_by_author	PROCEDURE	Edit  Execute  Export  Drop

**Lab 4 :** Write a stored procedure that takes book\_id as an argument and returns the price of the book.

```
- DELIMITER //
CREATE PROCEDURE getprice_by_bookid (IN b_id int)
BEGIN
SELECT price FROM books WHERE book_id = b_id;
END//
DELIMITER ;
```

The screenshot shows the MySQL Workbench interface with the following details:

- Server:** 127.0.0.1
- Database:** library\_db
- Execution Results:**
  - Your SQL query has been executed successfully.
  - 1 row affected by the last statement inside the procedure.
  - SQL Query: `SET @p0='3'; CALL `getprice_by_bookid`(@p0);`
  - Execution results of routine `getprice\_by\_bookid`:

price
90
- Routines:**
  - Check all, Export, Drop buttons.
  - Table:

Name	Type	Returns
getbook_by_author	PROCEDURE	<a href="#">Edit</a> <a href="#">Execute</a> <a href="#">Export</a> <a href="#">Drop</a>
getprice_by_bookid	PROCEDURE	<a href="#">Edit</a> <a href="#">Execute</a> <a href="#">Export</a> <a href="#">Drop</a>

## 17. SQL View

**Lab 3 :** Create a view to show only the title, author, and price of books from the books table.

- CREATE VIEW book\_view AS SELECT title,author,price FROM books

The screenshot shows the phpMyAdmin interface for a database named 'library\_db'. A warning message at the top states: 'Current selection does not contain a unique column. Grid edit, Edit, Copy and Delete features may result in undesired behavior.' Below this, a green status bar indicates: 'Showing rows 0 - 4 (5 total, Query took 0.0034 seconds.)'. The SQL query listed is: 'SELECT \* FROM `book\_view`'. The results grid displays the following data:

	title	author	price
<input type="checkbox"/>	The Alchemist	Paulo Coelho	60
<input type="checkbox"/>	Clean Code	Robert C. Martin	550
<input type="checkbox"/>	Atomic Habits	James Clear	90
<input type="checkbox"/>	Wings of Fire	A. P. J. Abdul Kalam	95
<input type="checkbox"/>	Think and Grow Rich	Napoleon Hill	400

**Lab 4 :** Create a view to display members who joined before 2020.

- CREATE VIEW member\_view\_before\_2020 AS SELECT \* FROM members WHERE date\_of\_membership < '2020-01-01'

The screenshot shows the phpMyAdmin interface for a database named 'library\_db'. A warning message at the top states: 'Current selection does not contain a unique column. Grid edit, Edit, Copy and Delete features may result in undesired behavior.' Below this, a green status bar indicates: 'Showing rows 0 - 1 (2 total, Query took 0.0005 seconds.)'. The SQL query listed is: 'SELECT \* FROM `member\_view\_before\_2020`'. The results grid displays the following data:

	member_id	member_name	date_of_membership	email
<input type="checkbox"/>	5	Vishal Kumar	2019-05-20	vishal@gmail.com
<input type="checkbox"/>	6	Jeel Kumar	2015-02-10	jeel@gmail.com

## 18. SQL Trigger

**Lab 3 :** Create a trigger to automatically update the last\_modified timestamp of the books table whenever a record is updated.

```
- DELIMITER $$  
CREATE TRIGGER trg_update_books_timestamp  
BEFORE UPDATE ON books  
FOR EACH ROW  
BEGIN  
    SET NEW.last_modified = CURRENT_TIMESTAMP;  
END$$  
DELIMITER ;
```

The screenshot shows the phpMyAdmin interface for the 'books' table in the 'library\_db' database. The table has columns: book\_id, title, and last\_modified. There are 7 rows of data. The last row, book\_id 7, has a timestamp of 2025-12-27 11:20:13. The previous row, book\_id 6, also has the same timestamp. A recent update was made to book\_id 2, changing its price to 650. The 'last\_modified' column for book\_id 2 now shows the current timestamp, 2025-12-27 11:23:33.

book_id	title	last_modified
1	The Alchemist	2025-12-27 11:20:13
2	Clean Code	2025-12-27 11:23:33
3	Atomic Habits	2025-12-27 11:20:13
4	Wings of Fire	2025-12-27 11:20:13
5	Think and Grow Rich	2025-12-27 11:20:13
6	Data Structures	2025-12-27 11:20:13
7	Operating System Con	2025-12-27 11:20:13

**Lab 4 :** Create a trigger that inserts a log entry into a log\_changes table whenever a DELETE operation is performed on the books table.

- DELIMITER \$\$

```
CREATE TRIGGER trg_books_delete_log
AFTER DELETE ON books
FOR EACH ROW
BEGIN
    INSERT INTO log_changes (book_id, action_type)
    VALUES (OLD.book_id, 'DELETE');
END$$
```

DELIMITER ;

The screenshot shows the phpMyAdmin interface for a MySQL database named 'library\_db'. The current table is 'log\_changes'. The top navigation bar includes tabs for 'Browse', 'Structure', 'SQL', 'Search', 'Insert', 'Export', 'Import', and 'Privileges'. Below the tabs, there's a 'Show query box' button. The main area displays two messages: a green success message '1 row deleted. (Query took 0.0004 seconds.)' and a green success message 'Showing rows 0 - 0 (1 total, Query took 0.0003 seconds.)'. Between these messages are the SQL queries: 'DELETE FROM books WHERE book\_id = 7;' and 'SELECT \* FROM log\_changes;'. Below the messages are buttons for 'Profiling', 'Edit inline', 'Edit', 'Explain SQL', 'Create PHP code', and 'Refresh'. At the bottom, there are buttons for 'Show all', 'Number of rows: 25', 'Filter rows: Search this table', and 'Extra options'. The 'log\_changes' table is shown with the following data:

	log_id	book_id	action_type	action_time
<input type="checkbox"/>	1	7	DELETE	2025-12-27 11:27:34

Below the table are buttons for 'Edit', 'Copy', and 'Delete'.

## 19. Introduction to PL/SQL

**Lab 3 :** Write a PL/SQL block to insert a new book into the books table and display a confirmation message.

```
-  DELIMITER $$  
CREATE PROCEDURE insert_book()  
BEGIN  
    INSERT INTO books  
    (book_id, title, author, publisher, year_of_publication, price, genre)  
    VALUES (10, 'MySQL Basics', 'John Smith', 'Pearson', 2023, 450, 'Database');  
    SELECT 'Book inserted successfully' AS message;  
END$$  
DELIMITER ;
```

The screenshot shows the MySQL Workbench interface with the following details:

- Server:** 127.0.0.1
- Database:** library\_db
- Table:** books
- Toolbar:** Browse, Structure, SQL, Search, Insert, Export
- Show query box:** A button labeled "Show query box".
- Warning Message:** "Current selection does not contain a unique column. Grid edit, checkbox, Edit, Copy and Delete operations are disabled." (partially visible).
- Status Bar:** "Showing rows 0 - 0 (1 total, Query took 0.0143 seconds.)"
- Query Result:** "CALL insert\_book();"
- Action Buttons:** [Edit inline], [Edit], [Create PHP code]
- Filter Options:** Show all, Number of rows: 25, Filter rows: Search this table
- Extra Options:** A button labeled "Extra options".
- Message Area:** "message" followed by the confirmation message "Book inserted successfully".

**Lab 4 :** Write a PL/SQL block to display the total number of books in the books table.

- DELIMITER \$\$

```
CREATE PROCEDURE total_books_count()
BEGIN
    DECLARE total INT;
    SELECT COUNT(*) INTO total FROM books;
    SELECT total AS total_books;
END$$
```

DELIMITER ;

The screenshot shows the phpMyAdmin interface for a database named 'library\_db' with a table named 'books'. The top navigation bar includes 'Server: 127.0.0.1', 'Database: library\_db', 'Table: books', and tabs for 'Browse', 'Structure', 'SQL', 'Search', 'Insert', 'Export', and 'Import'. A 'Show query box' button is visible. A warning message states: 'Current selection does not contain a unique column. Grid edit, checkbox, Edit, Copy and Delete features are disabled.' Below this, a success message says: 'Showing rows 0 - 0 (1 total, Query took 0.0006 seconds.)'. The SQL query shown is 'call total\_books\_count();'. There are buttons for '[Edit inline]', '[Edit]', and '[Create PHP code]'. At the bottom, there are options to 'Show all' (unchecked), set 'Number of rows' to 25, and a 'Filter rows' search bar. A link 'Extra options' is also present. The result table shows one row with the value '8' under the column 'total\_books'.

total_books
8

## 20. PL/SQL Syntax

**Lab 3 :** Write a PL/SQL block to declare variables for book\_id and price, assign values, and display the results.

```
-  DELIMITER $$  
CREATE PROCEDURE show_book_variables()  
BEGIN  
    DECLARE v_book_id INT;  
    DECLARE v_price DECIMAL(10,2);  
    SET v_book_id = 2;  
    SET v_price = 550.00;  
    SELECT v_book_id AS book_id, v_price AS price;  
END$$  
DELIMITER ;
```

The screenshot shows the phpMyAdmin interface with the following details:

- Server:** 127.0.0.1
- Database:** library\_db
- Table:** books
- Toolbar:** Browse, Structure, SQL, Search, Insert, Export, Import
- Show query box:** A button labeled "Show query box".
- Warning message:** "Current selection does not contain a unique column. Grid edit, checkbox, Edit, Copy and Delete features are disabled." (with a yellow warning icon).
- Success message:** "Showing rows 0 - 0 (1 total, Query took 0.0005 seconds.)" (with a green checkmark icon).
- Query result:** `CALL show_book_variables();`
- Action buttons:** [Edit inline] [Edit] [Create PHP code]
- Filter options:** Show all (unchecked), Number of rows: 25, Filter rows: Search this table.
- Extra options:** A button labeled "Extra options".
- Data grid:** A table with columns `book_id` and `price`. One row is displayed: `2 550.00`.

**Lab 4** : Write a PL/SQL block using constants and perform arithmetic operations on book prices.

- DELIMITER \$\$

```
CREATE PROCEDURE book_price_calculation()
BEGIN
    DECLARE base_price DECIMAL(10,2) DEFAULT 500;
    DECLARE gst_rate DECIMAL(10,2) DEFAULT 18;
    DECLARE gst_amount DECIMAL(10,2);
    DECLARE final_price DECIMAL(10,2);
    SET gst_amount = (base_price * gst_rate) / 100;
    SET final_price = base_price + gst_amount;
    SELECT
        base_price AS base_price,
        gst_amount AS gst_amount,
        final_price AS final_price;
END$$
```

DELIMITER ;

The screenshot shows the phpMyAdmin interface with the following details:

- Server:** 127.0.0.1
- Database:** library\_db
- Table:** books
- Toolbar:** Browse, Structure, SQL, Search, Insert, Export, Import
- Message Bar:** Shows a warning: "Current selection does not contain a unique column. Grid edit, checkbox, Edit, Copy and Delete fea".
- Status Bar:** Shows "Showing rows 0 - 0 (1 total, Query took 0.0005 seconds.)".
- Query Result:** A single row returned by the query:

base_price	gst_amount	final_price
500.00	90.00	590.00
- Action Buttons:** [Edit inline], [Edit], [Create PHP code]
- Filter Options:** Show all, Number of rows: 25, Filter rows: Search this table
- Extra Options:** Extra options button

## 21. PL/SQL Control Structures

**Lab 3 :** Write a PL/SQL block using IF-THEN-ELSE to check if a book's price is above \$100 and print a message accordingly.

```
- DELIMITER $$  
CREATE PROCEDURE check_book_price()  
BEGIN  
    DECLARE v_price DECIMAL(10,2);  
    SET v_price = 150;  
    IF v_price > 100 THEN  
        SELECT 'Book price is above $100' AS message;  
    ELSE  
        SELECT 'Book price is $100 or below' AS message;  
    END IF;  
END$$  
DELIMITER ;
```

The screenshot shows the MySQL Workbench interface with the following details:

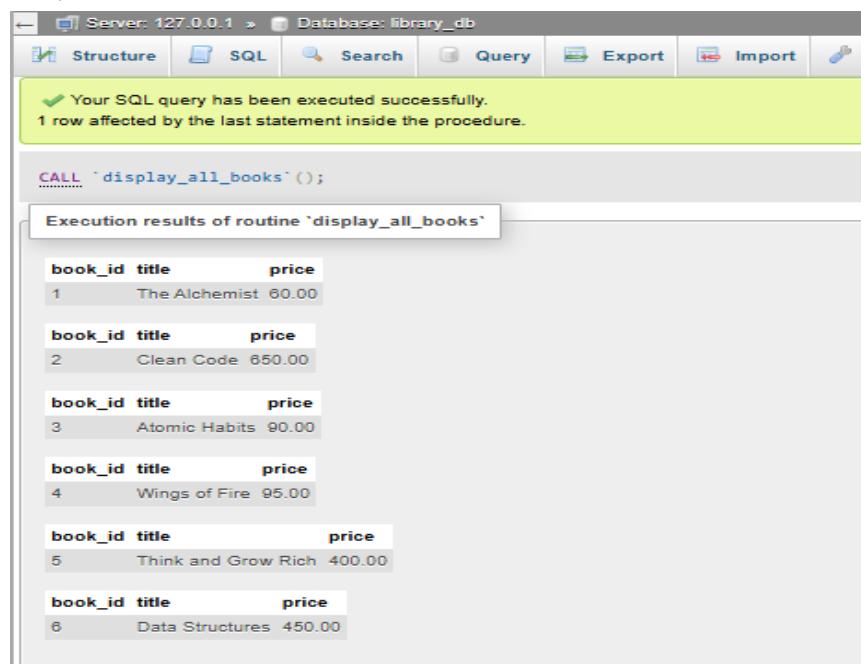
- Server:** 127.0.0.1
- Database:** library\_db
- Toolbar:** Structure, SQL, Search, Query, Export, Import
- Show query box:** A button labeled "Show query box".
- Message Bar:** A warning message: "⚠ Current selection does not contain a unique column. Grid edit, checkbox, Edit, Copy are disabled." (translated from Spanish).
- Status Bar:** "Showing rows 0 - 0 (1 total, Query took 0.0004 seconds.)"
- Query Editor:** The SQL code: "CALL check\_book\_price();". Below it are buttons for "[Edit inline]", "[Edit]", and "[Create PHP code]".
- Result Set:** A single row with the message "Book price is above \$100".
- Table Row Headers:** "message" (highlighted in blue), "Book price is above \$100".
- Bottom Navigation:** Buttons for "Show all" (unchecked), "Number of rows:" (set to 25), "Filter rows:", and a search bar "Search this table".
- Extra Options:** A button labeled "Extra options".

**Lab 4 :** Use a FOR LOOP in PL/SQL to display the details of all books one by one.

- DELIMITER \$\$

```
CREATE PROCEDURE display_all_books()
BEGIN
    DECLARE done INT DEFAULT 0;
    DECLARE v_id INT;
    DECLARE v_title VARCHAR(100);
    DECLARE v_price DECIMAL(10,2);
    DECLARE book_cursor CURSOR FOR
        SELECT book_id, title, price FROM books;
    DECLARE CONTINUE HANDLER FOR NOT FOUND SET done = 1;
    OPEN book_cursor;
    read_loop: LOOP
        FETCH book_cursor INTO v_id, v_title, v_price;
        IF done = 1 THEN
            LEAVE read_loop;
        END IF;
        SELECT v_id AS book_id, v_title AS title, v_price AS price;
    END LOOP;
    CLOSE book_cursor;
END$$
```

DELIMITER ;



The screenshot shows the MySQL Workbench interface with the following details:

- Server:** 127.0.0.1
- Database:** library\_db
- Execution results of routine 'display\_all\_books':**

book_id	title	price
1	The Alchemist	60.00
2	Clean Code	650.00
3	Atomic Habits	90.00
4	Wings of Fire	95.00
5	Think and Grow Rich	400.00
6	Data Structures	450.00

## 22. SQL Cursors

**Lab 3 :** Write a PL/SQL block using an explicit cursor to fetch and display all records from the members table.

```
-  DELIMITER //
CREATE PROCEDURE fetch_all_members()
BEGIN
    DECLARE v_member_id INT;
    DECLARE v_member_name VARCHAR(100);
    DECLARE v_date_of_membership DATE;
    DECLARE v_email VARCHAR(100);
    DECLARE done INT DEFAULT 0;
    DECLARE member_cursor CURSOR FOR
        SELECT member_id, member_name, date_of_membership, email FROM
members;
    DECLARE CONTINUE HANDLER FOR NOT FOUND SET done = 1;
    OPEN member_cursor;
    read_loop: LOOP
        FETCH member_cursor INTO v_member_id, v_member_name,
v_date_of_membership, v_email;
        IF done = 1 THEN
            LEAVE read_loop;
        END IF;
        SELECT v_member_id AS MemberID,
            v_member_name AS Name,
            v_date_of_membership AS MembershipDate,
            v_email AS Email;
    END LOOP;
    CLOSE member_cursor;
END;
// 
DELIMITER ;
```

Your SQL query has been executed successfully.  
1 row affected by the last statement inside the procedure.

```
CALL `fetch_all_members`();
```

**Execution results of routine 'fetch\_all\_members'**

MemberID	Name	MembershipDate	Email
1	Ayush Patel	2024-01-15	ayush@gmail.com
4	Dev Joshi	2021-04-12	dev@gmail.com
5	Vishal Kumar	2019-05-20	vishal@gmail.com
6	Jeel Kumar	2015-02-10	jeel@gmail.com

**Lab 4 :** Create a cursor to retrieve books by a particular author and display their titles.

- DELIMITER //

```

CREATE PROCEDURE fetch_books_by_author(IN author_name VARCHAR(100))
BEGIN
    DECLARE v_title VARCHAR(255);
    DECLARE done INT DEFAULT 0;
    DECLARE books_cursor CURSOR FOR
        SELECT title FROM books WHERE author = author_name;
    DECLARE CONTINUE HANDLER FOR NOT FOUND SET done = 1;
    OPEN books_cursor;
    read_loop: LOOP
        FETCH books_cursor INTO v_title;
        IF done = 1 THEN
            LEAVE read_loop;
        END IF;
        SELECT v_title AS BookTitle;
    END LOOP;
    CLOSE books_cursor;
END;
//
```

DELIMITER ;

The screenshot shows the MySQL Workbench interface with the database set to 'library\_db'. A green message bar at the top indicates that the SQL query has been executed successfully and one row was affected by the last statement inside the procedure. Below this, the SQL query `SET @p0='Paulo Coelho'; CALL `fetch\_books\_by\_author`(@p0);` is shown. A results pane titled 'Execution results of routine 'fetch\_books\_by\_author'' displays a single row with the book title 'The Alchemist'.

```

SET @p0='Paulo Coelho'; CALL `fetch_books_by_author`(@p0);

Execution results of routine 'fetch_books_by_author'

BookTitle
The Alchemist

```

## 23. Rollback and Commit Savepoint

**Lab 3 :** Perform a transaction that includes inserting a new member, setting a SAVEPOINT, and rolling back to the savepoint after making updates.

- START TRANSACTION;
- INSERT INTO members (member\_name,date\_of\_membership,email)
- VALUES ('Rahul Sharma', '2025-12-27', 'rahul@example.com');
- SAVEPOINT before\_update;
- UPDATE members
- SET email = 'rahul\_new@example.com'
- WHERE member\_id = 7;
- ROLLBACK TO SAVEPOINT before\_update;
- COMMIT;

The screenshot shows the MySQL Workbench interface with the database set to 'library\_db' and the table 'members'. A green message bar at the top indicates that rows 0 - 4 (5 total) were shown, and the query took 0.0003 seconds. Below this, the SQL query `SELECT \* FROM `members`` is shown. The results pane displays the following data:

member_id	member_name	date_of_membership	email
1	Ayush Patel	2024-01-15	ayush@gmail.com
4	Dev Joshi	2021-04-12	dev@gmail.com
5	Vishal Kumar	2019-05-20	vishal@gmail.com
6	Jeel Kumar	2015-02-10	jeel@gmail.com
7	Rahul Sharma	2025-12-27	rahul@example.com

**Lab 4 :** Use COMMIT after successfully inserting multiple books into the books table, then use ROLLBACK to undo a set of changes made after a savepoint.

- START TRANSACTION;  
INSERT INTO books (title, author, publisher, year\_of\_publication, price, genre)  
VALUES  
('Effective Java', 'Joshua Bloch', 'Addison-Wesley', 2018, 750, 'Programming'),  
('You Don't Know JS', 'Kyle Simpson', 'O'Reilly Media', 2015, 680, 'Programming');  
COMMIT;
- START TRANSACTION;  
SAVEPOINT sp\_books;  
UPDATE books  
SET price = price + 50  
WHERE genre = 'Programming';  
DELETE FROM books  
WHERE title = 'Atomic Habits';  
ROLLBACK TO sp\_books;  
COMMIT;

The screenshot shows the phpMyAdmin interface for a MySQL database named 'library\_db'. The current table is 'books'. The top navigation bar includes tabs for 'Browse', 'Structure', 'SQL', 'Search', 'Insert', 'Export', 'Import', 'Privileges', 'Operations', 'Tracking', and 'Triggers'. Below the tabs, a message indicates 'Showing rows 0 - 8 (9 total, Query took 0.0004 seconds.)'. The SQL query shown is 'SELECT \* FROM `books`'. The results table displays the following data:

book_id	title	author	publisher	year_of_publication	price	genre	last_modified
1	The Alchemist	Paulo Coelho	HarperCollins	1988	60	Fiction / Philosophy	2025-12-27 11:20:13
2	Clean Code	Robert C. Martin	Pearson	2008	700	Programming	2025-12-27 13:47:30
4	Wings of Fire	A. P. J. Abdul Kalam	Universities Press	1999	95	Biography	2025-12-27 11:20:13
5	Think and Grow Rich	Napoleon Hill	Rupa	1937	400	Self Help	2025-12-27 11:20:13
6	Data Structures	Mark Allen Weiss	Pearson	2014	500	Programming	2025-12-27 13:47:30
7	Effective Java	Joshua Bloch	Addison-Wesley	2018	800	Programming	2025-12-27 13:47:30
8	You Don't Know JS	Kyle Simpson	O'Reilly Media	2015	730	Programming	2025-12-27 13:47:30