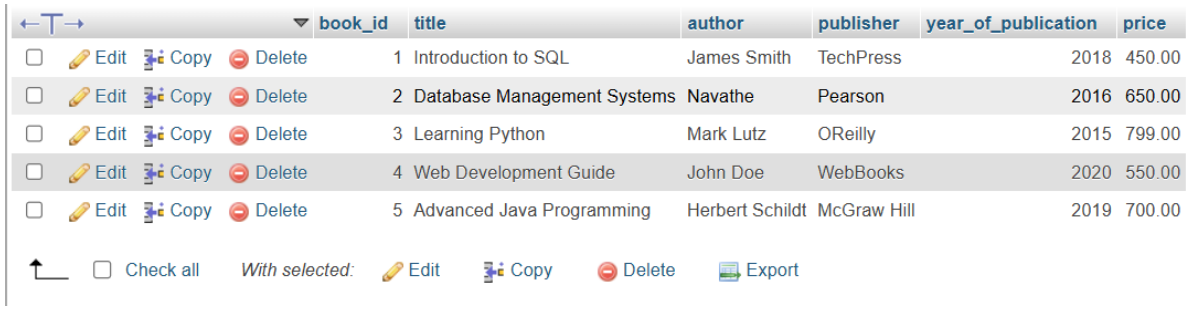
QUESTION:

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.

CODE:

```
CREATE DATABASE library_db;

CREATE TABLE books (
    book_id INT PRIMARY KEY,
    title VARCHAR(100),
    author VARCHAR(100),
    publisher VARCHAR(100),
    year_of_publication INT,
    price DECIMAL(10,2)
);

INSERT INTO books (book_id, title, author, publisher, year_of_publication, price)
VALUES
(1, 'Introduction to SQL', 'James Smith', 'TechPress', 2018, 450.00),
(2, 'Database Management Systems', 'Navathe', 'Pearson', 2016, 650.00),
(3, 'Learning Python', 'Mark Lutz', 'OReilly', 2015, 799.00),
(4, 'Web Development Guide', 'John Doe', 'WebBooks', 2020, 550.00),
(5, 'Advanced Java Programming', 'Herbert Schildt', 'McGraw Hill', 2019, 700.00);
```

SCREENSHOT:



	book_id	title	author	publisher	year_of_publication	price
Edit Copy Delete	1	Introduction to SQL	James Smith	TechPress	2018	450.00
Edit Copy Delete	2	Database Management Systems	Navathe	Pearson	2016	650.00
Edit Copy Delete	3	Learning Python	Mark Lutz	OReilly	2015	799.00
Edit Copy Delete	4	Web Development Guide	John Doe	WebBooks	2020	550.00
Edit Copy Delete	5	Advanced Java Programming	Herbert Schildt	McGraw Hill	2019	700.00

Check all *With selected:* Edit Copy Delete Export

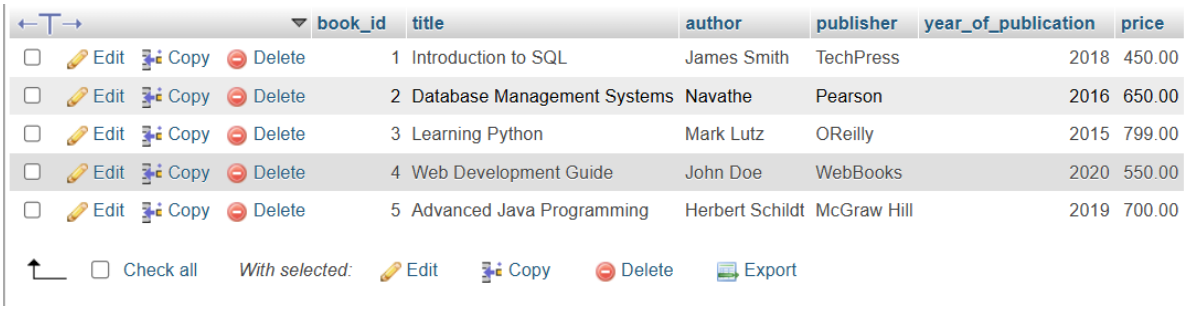
QUESTION:

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.

CODE:

```
CREATE TABLE members (
    member_id INT PRIMARY KEY,
    member_name VARCHAR(100),
    date_of_membership DATE,
    email VARCHAR(100)
);
INSERT INTO members (member_id, member_name, date_of_membership, email)
VALUES
(1, 'Rahul Sharma', '2022-01-15', 'rahul.sharma@example.com'),
(2, 'Priya Patel', '2021-12-20', 'priya.patel@example.com'),
(3, 'Amit Kumar', '2023-03-05', 'amit.kumar@example.com'),
(4, 'Sneha Mehta', '2022-07-10', 'sneha.mehta@example.com'),
(5, 'Karan Joshi', '2023-01-25', 'karan.joshi@example.com');
```

SCREENSHOT:



	book_id	title	author	publisher	year_of_publication	price
Edit Copy Delete	1	Introduction to SQL	James Smith	TechPress	2018	450.00
Edit Copy Delete	2	Database Management Systems	Navathe	Pearson	2016	650.00
Edit Copy Delete	3	Learning Python	Mark Lutz	OReilly	2015	799.00
Edit Copy Delete	4	Web Development Guide	John Doe	WebBooks	2020	550.00
Edit Copy Delete	5	Advanced Java Programming	Herbert Schildt	McGraw Hill	2019	700.00

Check all *With selected:* Edit Copy Delete Export

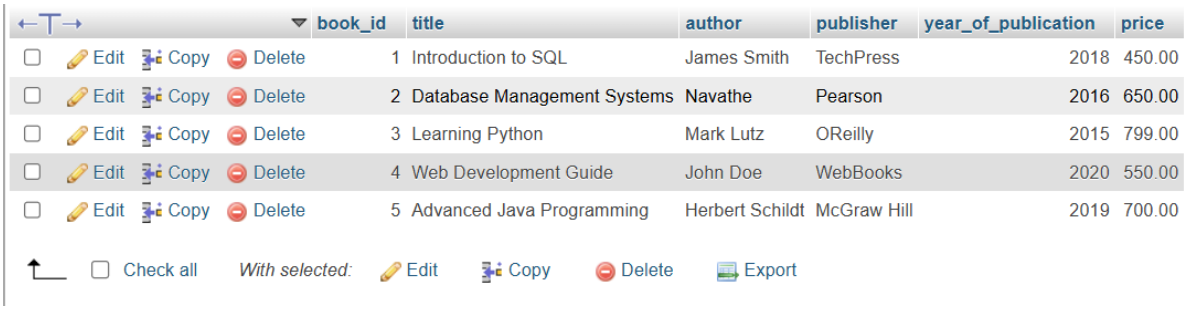
QUESTION:

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

CODE:

```
SELECT *

FROM members

WHERE date_of_membership < '2022-01-01'

ORDER BY date_of_membership ASC;
```

SCREENSHOT:



	book_id	title	author	publisher	year_of_publication	price
Edit Copy Delete	1	Introduction to SQL	James Smith	TechPress	2018	450.00
Edit Copy Delete	2	Database Management Systems	Navathe	Pearson	2016	650.00
Edit Copy Delete	3	Learning Python	Mark Lutz	OReilly	2015	799.00
Edit Copy Delete	4	Web Development Guide	John Doe	WebBooks	2020	550.00
Edit Copy Delete	5	Advanced Java Programming	Herbert Schildt	McGraw Hill	2019	700.00

Check all *With selected:* Edit Copy Delete Export

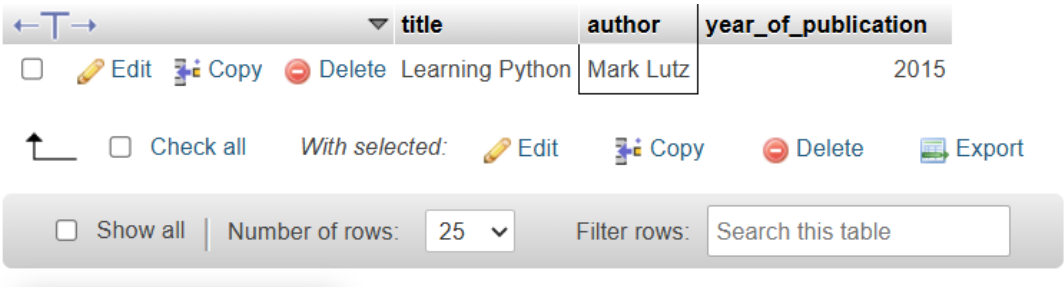
QUESTION:

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.

CODE:

```
SELECT title, author, year_of_publication

FROM books

WHERE author = 'Mark Lutz'

ORDER BY year_of_publication DESC;
```

SCREENSHOT:



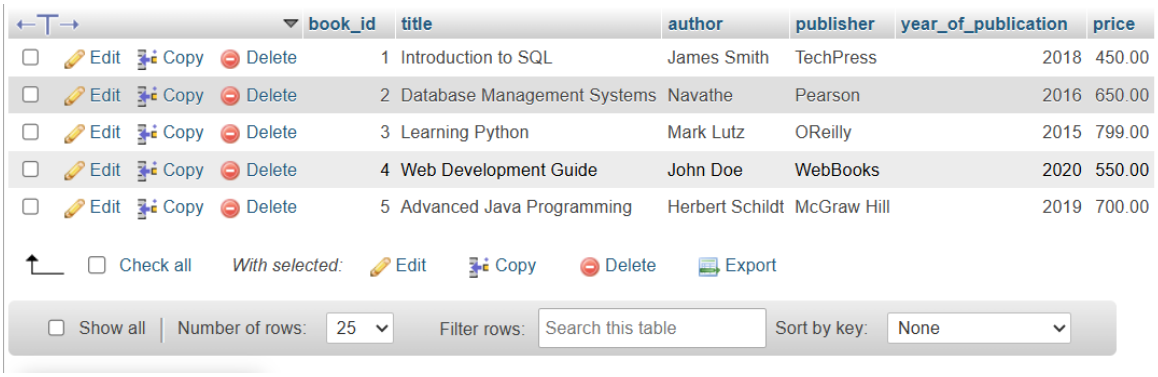
QUESTION:

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

CODE:

```
ALTER TABLE books

ADD CONSTRAINT chk_price CHECK (price > 0);
```

SCREENSHOT:



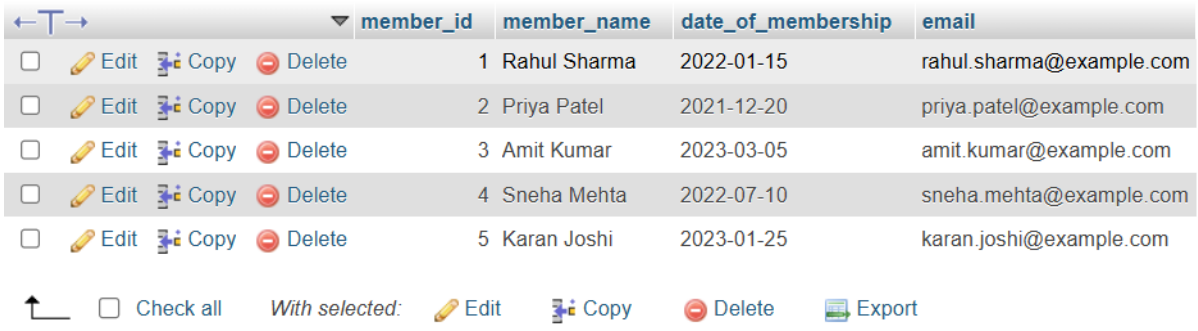
QUESTION:

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

CODE:

```
ALTER TABLE members

ADD CONSTRAINT unique_email UNIQUE (email);
```

SCREENSHOT:



	member_id	member_name	date_of_membership	email
Edit Copy Delete	1	Rahul Sharma	2022-01-15	rahul.sharma@example.com
Edit Copy Delete	2	Priya Patel	2021-12-20	priya.patel@example.com
Edit Copy Delete	3	Amit Kumar	2023-03-05	amit.kumar@example.com
Edit Copy Delete	4	Sneha Mehta	2022-07-10	sneha.mehta@example.com
Edit Copy Delete	5	Karan Joshi	2023-01-25	karan.joshi@example.com

Check all *With selected:* Edit Copy Delete Export

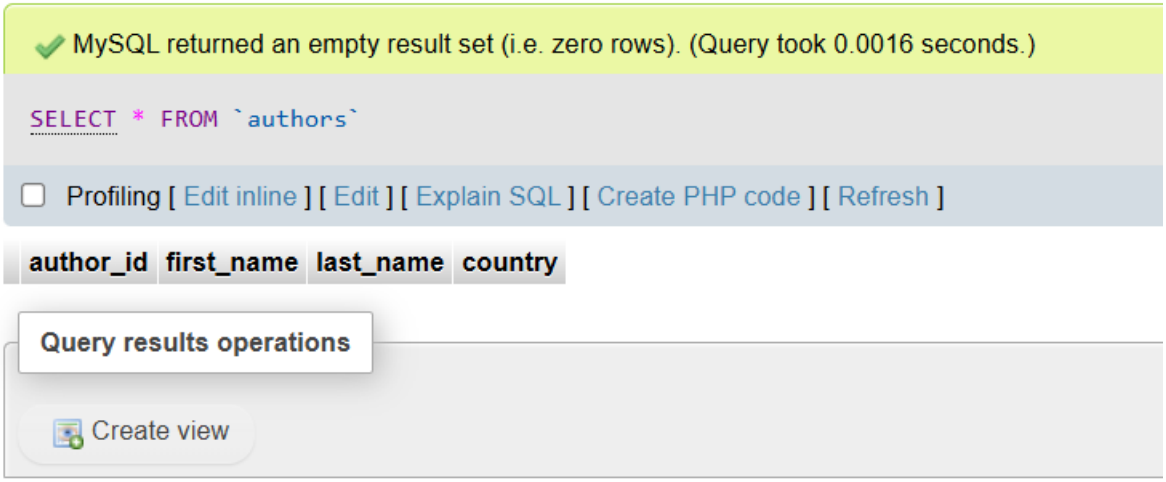
QUESTION:

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.

CODE:

```
CREATE TABLE authors (
   author_id INT PRIMARY KEY,
   first_name VARCHAR(100),
   last_name VARCHAR(100),
   country VARCHAR(100)
);
```

SCREENSHOT:



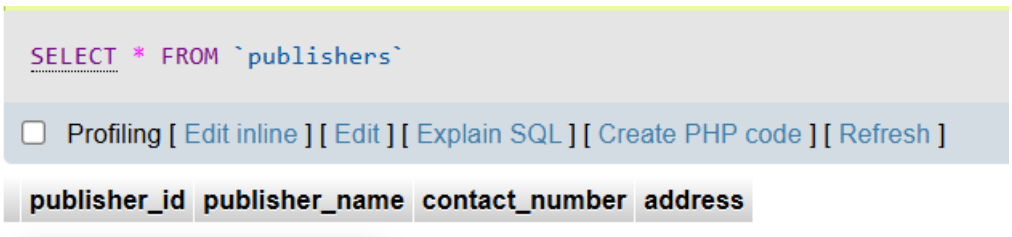
QUESTION:

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.

CODE:

```
CREATE TABLE publishers (
    publisher_id INT PRIMARY KEY,
    publisher_name VARCHAR(150),
    contact_number VARCHAR(20) UNIQUE,
    address VARCHAR(200)
);
```

SCREENSHOT:



QUESTION:

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

CODE:

```
ALTER TABLE books
ADD genre VARCHAR(50);
UPDATE books
SET genre = 'Education'
WHERE book_id = 1;

UPDATE books
SET genre = 'Technology'
WHERE book_id = 2;

UPDATE books
SET genre = 'Programming'
WHERE book_id = 3;

UPDATE books
SET genre = 'Web Development'
WHERE book_id = 4;

UPDATE books
SET genre = 'Java'
WHERE book_id = 5;
```

SCREENSHOT:

	book_id	title	author	publisher	year_of_publication	price	genre
Edit Copy Delete	1	Introduction to SQL	James Smith	TechPress	2018	450.00	Education
Edit Copy Delete	2	Database Management Systems	Navathe	Pearson	2016	650.00	Technology
Edit Copy Delete	3	Learning Python	Mark Lutz	OReilly	2015	799.00	Programming
Edit Copy Delete	4	Web Development Guide	John Doe	WebBooks	2020	550.00	Web Development
Edit Copy Delete	5	Advanced Java Programming	Herbert Schildt	McGraw Hill	2019	700.00	Java

Check all *With selected:* Edit Copy Delete Export

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

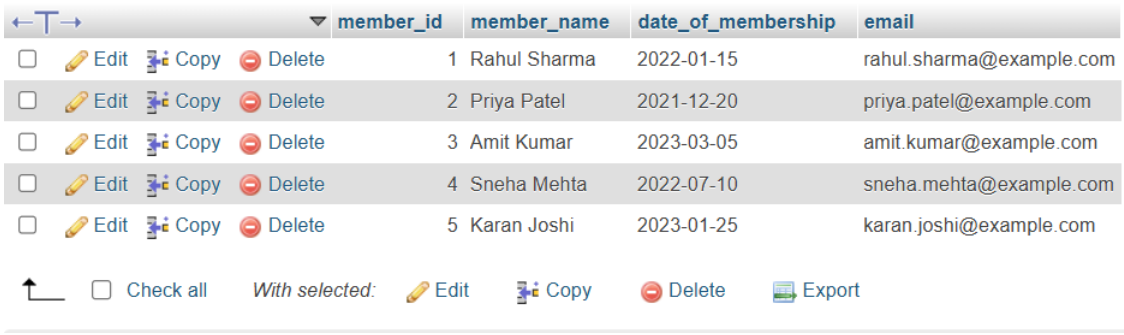
QUESTION:

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

CODE:

```
ALTER TABLE members

MODIFY email VARCHAR(100);
```

SCREENSHOT:



	member_id	member_name	date_of_membership	email
Edit Copy Delete	1	Rahul Sharma	2022-01-15	rahul.sharma@example.com
Edit Copy Delete	2	Priya Patel	2021-12-20	priya.patel@example.com
Edit Copy Delete	3	Amit Kumar	2023-03-05	amit.kumar@example.com
Edit Copy Delete	4	Sneha Mehta	2022-07-10	sneha.mehta@example.com
Edit Copy Delete	5	Karan Joshi	2023-01-25	karan.joshi@example.com

Check all *With selected:* Edit Copy Delete Export

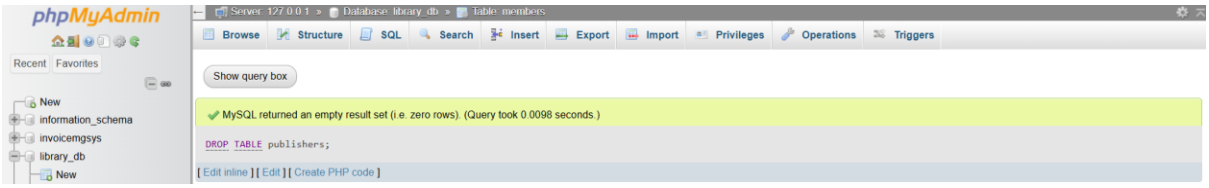
QUESTION:

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

CODE:

```
DROP TABLE publishers;
```

SCREENSHOT:



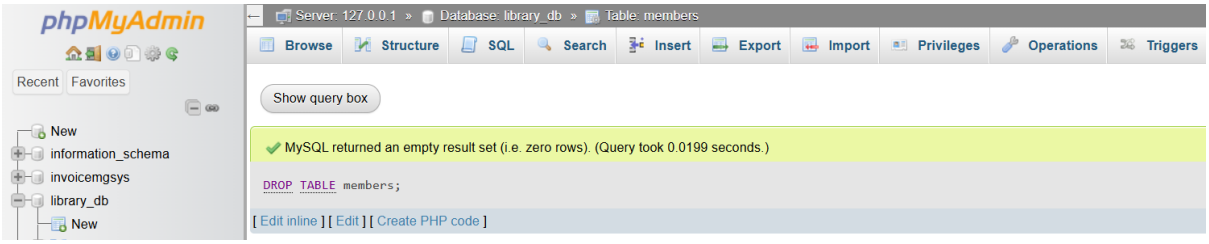
QUESTION:

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

CODE:

```
CREATE TABLE members_backup AS

SELECT * FROM members;

DROP TABLE members;
```

SCREENSHOT:



QUESTION:

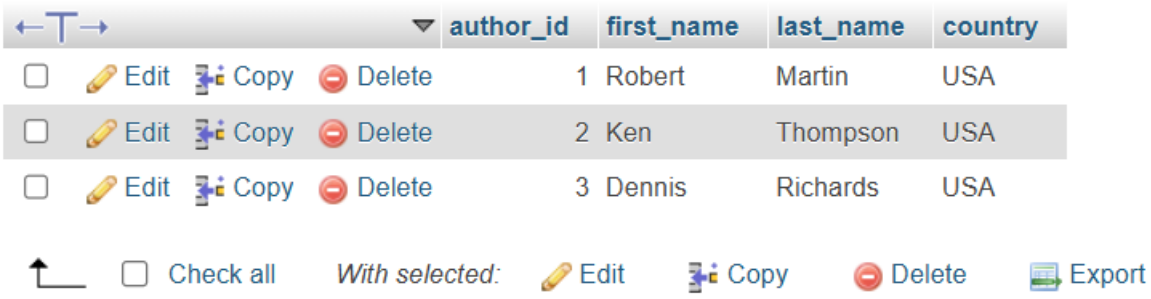
Lab 4: Insert three new authors into the authors table, then update the last name of one of the authors.

CODE:

```
INSERT INTO authors (author_id, first_name, last_name, country)

VALUES

(1, 'Robert', 'Martin', 'USA'),

(2, 'Ken', 'Thompson', 'USA'),

(3, 'Dennis', 'Ritchie', 'USA');
```

```
UPDATE authors

SET last_name = 'Richards'

WHERE author_id = 3;
```

SCREENSHOT:



	author_id	first_name	last_name	country
Edit Copy Delete	1	Robert	Martin	USA
Edit Copy Delete	2	Ken	Thompson	USA
Edit Copy Delete	3	Dennis	Richards	USA

Check all *With selected:* Edit Copy Delete Export

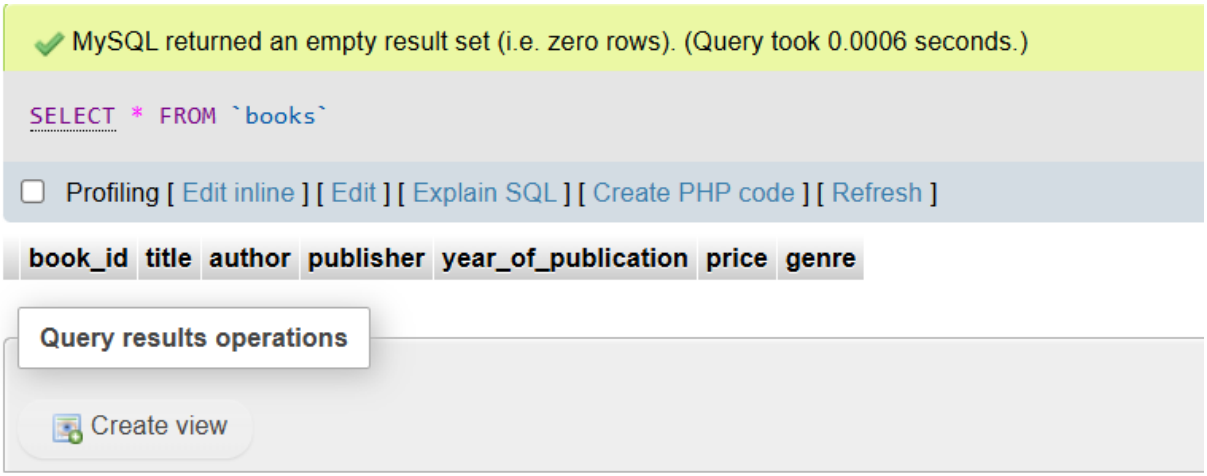
QUESTION:

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

CODE:

```
DELETE FROM books

WHERE price > 100;
```

SCREENSHOT:



MySQL returned an empty result set (i.e. zero rows). (Query took 0.0006 seconds.)

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

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

book_id	title	author	publisher	year_of_publication	price	genre

Query results operations

Create view

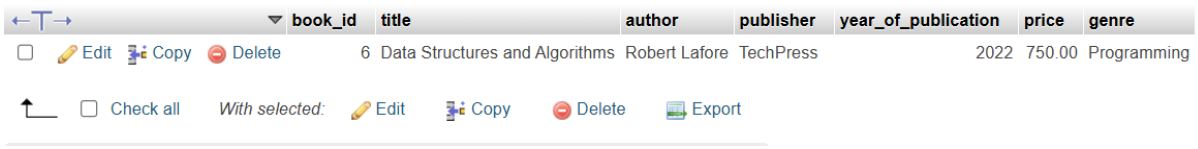
QUESTION:

Lab 3: Update the year_of_publication of a book with a specific book_id.

CODE:

```
UPDATE books

SET year_of_publication = 2021

WHERE book_id = 3;
```

SCREENSHOT:



	book_id	title	author	publisher	year_of_publication	price	genre
Edit Copy Delete	6	Data Structures and Algorithms	Robert Lafore	TechPress	2022	750.00	Programming

Check all With selected: Edit Copy Delete Export

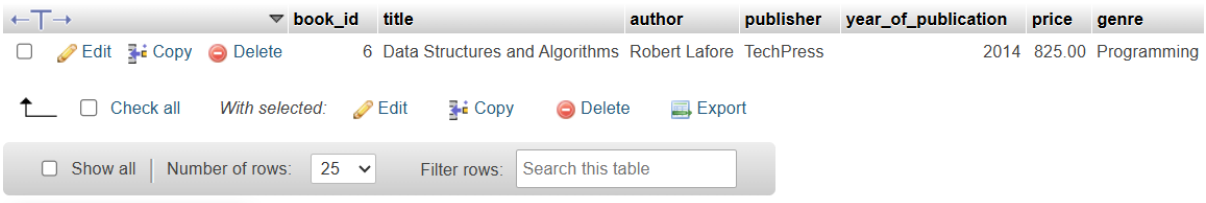
QUESTION:

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

CODE:

```
UPDATE books

SET price = price * 1.10

WHERE year_of_publication < 2015;
```

SCREENSHOT:



	book_id	title	author	publisher	year_of_publication	price	genre
Edit Copy Delete	6	Data Structures and Algorithms	Robert Lafore	TechPress	2014	825.00	Programming

Check all With selected: Edit Copy Delete Export

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

QUESTION:

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

CODE:

```
DELETE FROM members_backup

WHERE date_of_membership < '2020-01-01';
```

SCREENSHOT:

member_id	member_name	date_of_membership	email
1	Rahul Sharma	2022-01-15	rahul.sharma@example.com
2	Priya Patel	2021-12-20	priya.patel@example.com
3	Amit Kumar	2023-03-05	amit.kumar@example.com
4	Sneha Mehta	2022-07-10	sneha.mehta@example.com
5	Karan Joshi	2023-01-25	karan.joshi@example.com

QUESTION:

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

CODE:

```
DELETE FROM books

WHERE author IS NULL;
```

SCREENSHOT:

	book_id	title	author	publisher	year_of_publication	price	genre
Edit Copy Delete	6	Data Structures and Algorithms	Robert Lafore	TechPress	2014	825.00	Programming

Check all With selected: Edit Copy Delete Export

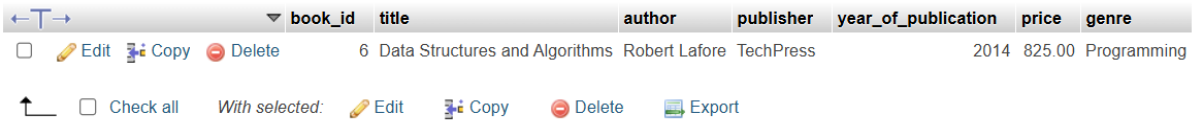
QUESTION:

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

CODE:

```
SELECT *

FROM books

WHERE price BETWEEN 50 AND 100;
```

SCREENSHOT:



	book_id	title	author	publisher	year_of_publication	price	genre
Edit Copy Delete	6	Data Structures and Algorithms	Robert Lafore	TechPress	2014	825.00	Programming

Check all With selected: Edit Copy Delete Export

QUESTION:

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

CODE:

```
SELECT *

FROM books

ORDER BY author ASC

LIMIT 3;
```

SCREENSHOT:

	book_id	title	author	publisher	year_of_publication	price	genre
Edit Copy Delete	6	Data Structures and Algorithms	Robert Lafore	TechPress	2014	825.00	Programming
Edit Copy Delete	7	Artificial Intelligence Basics	Stuart Russell	AI Press	2021	850.00	Technology
Edit Copy Delete	8	Machine Learning Guide	Tom Mitchell	ML Books	2019	950.00	Technology
Edit Copy Delete	9	Clean Code	Robert Martin	Prentice Hall	2008	700.00	Programming
Edit Copy Delete	10	Deep Learning	Ian Goodfellow	MIT Press	2016	1200.00	Technology

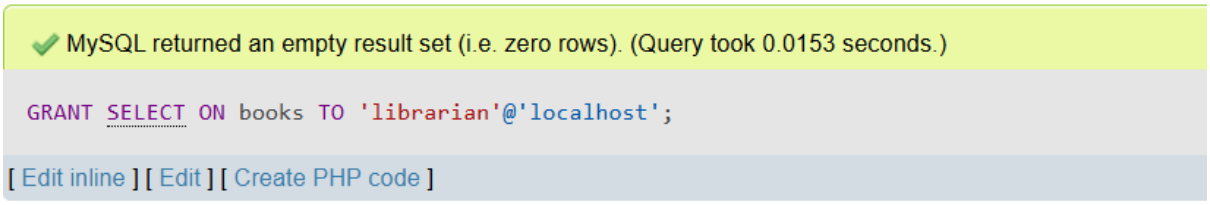
QUESTION:

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

CODE:

```
GRANT SELECT ON books TO 'librarian'@'localhost';
```

SCREENSHOT:



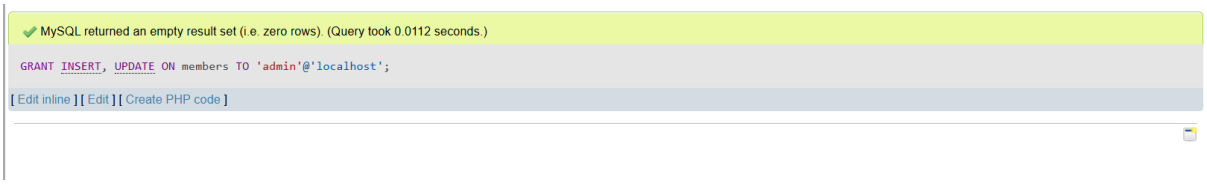
QUESTION:

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

CODE:

```
GRANT INSERT, UPDATE ON members TO 'admin'@'localhost';
```

SCREENSHOT:



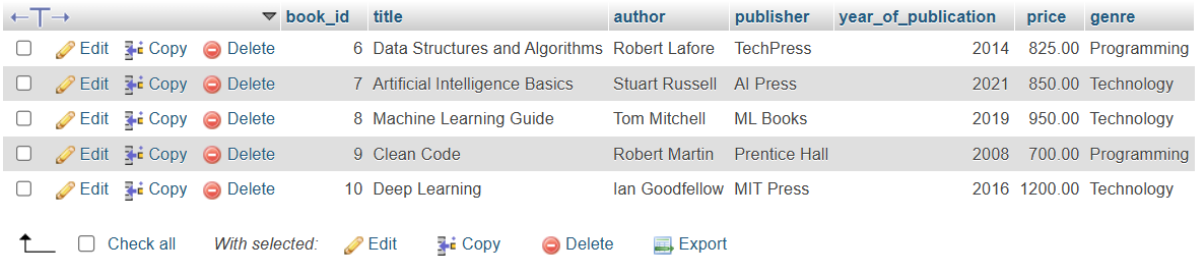
QUESTION:

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

CODE:

```
REVOKE INSERT ON books FROM 'librarian'@'localhost';
```

SCREENSHOT:



	book_id	title	author	publisher	year_of_publication	price	genre
Edit Copy Delete	6	Data Structures and Algorithms	Robert Lafore	TechPress	2014	825.00	Programming
Edit Copy Delete	7	Artificial Intelligence Basics	Stuart Russell	AI Press	2021	850.00	Technology
Edit Copy Delete	8	Machine Learning Guide	Tom Mitchell	ML Books	2019	950.00	Technology
Edit Copy Delete	9	Clean Code	Robert Martin	Prentice Hall	2008	700.00	Programming
Edit Copy Delete	10	Deep Learning	Ian Goodfellow	MIT Press	2016	1200.00	Technology

Check all *With selected:* Edit Copy Delete Export

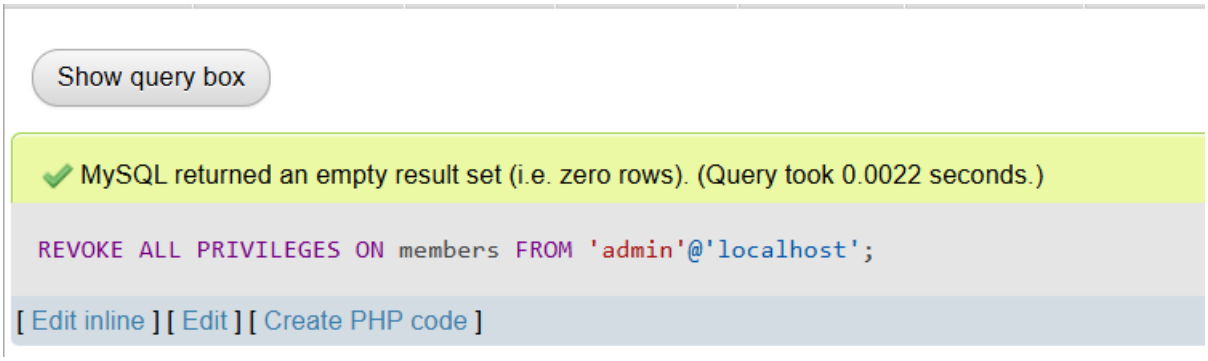
QUESTION:

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

CODE:

```
REVOKE ALL PRIVILEGES ON members FROM 'admin'@'localhost';
```

SCREENSHOT:



QUESTION:

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

CODE:

```
START TRANSACTION;

INSERT INTO books (book_id, title, author, publisher, year_of_publication, price, genre)

VALUES

(14, 'Cloud Computing Basics', 'Thomas Erl', 'TechWorld', 2020, 700.00, 'Technology');

ROLLBACK;
```

SCREENSHOT:

	book_id	title	author	publisher	year_of_publication	price	genre
Edit Copy Delete	6	Data Structures and Algorithms	Robert Lafore	TechPress	2014	825.00	Programming
Edit Copy Delete	7	Artificial Intelligence Basics	Stuart Russell	AI Press	2021	850.00	Technology
Edit Copy Delete	8	Machine Learning Guide	Tom Mitchell	ML Books	2019	950.00	Technology
Edit Copy Delete	9	Clean Code	Robert Martin	Prentice Hall	2008	700.00	Programming
Edit Copy Delete	10	Deep Learning	Ian Goodfellow	MIT Press	2016	1200.00	Technology
Edit Copy Delete	11	Networking Essentials	James Kurose	Pearson	2017	600.00	Technology
Edit Copy Delete	12	Operating Systems Concepts	Abraham Silberschatz	Wiley	2018	850.00	Technology
Edit Copy Delete	13	Computer Architecture	John Hennessy	Morgan Kaufmann	2015	900.00	Technology

QUESTION:

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

CODE:

```
START TRANSACTION;

SAVEPOINT sp1;

UPDATE members

SET member_name = 'Rahul S.'

WHERE member_id = 1;

UPDATE members

SET email = 'updated.priya@example.com'

WHERE member_id = 2;

ROLLBACK TO SAVEPOINT sp1;

COMMIT;
```

SCREENSHOT:

	member_id	member_name	date_of_membership	email
Edit Copy Delete	1	Rahul S.	2022-01-15	rahul.sharma@example.com
Edit Copy Delete	2	Priya Patel	2021-12-20	new.priya@example.com
Edit Copy Delete	3	Amit Kumar	2023-03-05	amit.kumar@example.com
Edit Copy Delete	4	Sneha Mehta	2022-07-10	sneha.mehta@example.com
Edit Copy Delete	5	Karan Joshi	2023-01-25	karan.joshi@example.com

Check all *With selected:* Edit Copy Delete Export

QUESTION:

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

CODE:

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

SCREENSHOT:

	member_id	member_name	date_of_membership	email
Edit Copy Delete	1	Rahul S.	2022-01-15	rahul.sharma@example.com
Edit Copy Delete	2	Priya Patel	2021-12-20	new.priya@example.com
Edit Copy Delete	3	Amit Kumar	2023-03-05	amit.kumar@example.com
Edit Copy Delete	4	Sneha Mehta	2022-07-10	sneha.mehta@example.com
Edit Copy Delete	5	Karan Joshi	2023-01-25	karan.joshi@example.com

Check all *With selected:* Edit Copy Delete Export

QUESTION:

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.

CODE:

```
SELECT
    books.title,
    authors.first_name,
    authors.last_name
FROM books
```

```
LEFT JOIN authors
    ON books.author = CONCAT(authors.first_name, ' ', authors.last_name)

UNION

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

SCREENSHOT:

title	first_name	last_name
Clean Code	Robert	Martin
Data Structures and Algorithms	*NULL*	*NULL*
Artificial Intelligence Basics	*NULL*	*NULL*
Machine Learning Guide	*NULL*	*NULL*
Deep Learning	*NULL*	*NULL*
Networking Essentials	*NULL*	*NULL*
Operating Systems Concepts	*NULL*	*NULL*
Computer Architecture	*NULL*	*NULL*
NULL	Ken	Thompson
NULL	Dennis	Richards

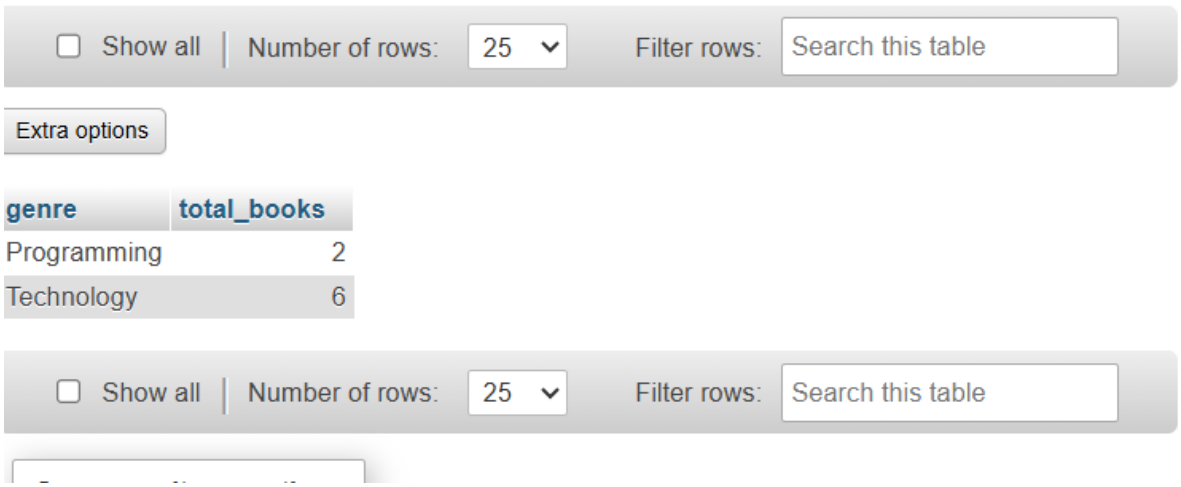
QUESTION:

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

CODE:

```
SELECT genre,COUNT(*) AS total_books

FROM books

GROUP BY genre;
```

SCREENSHOT:



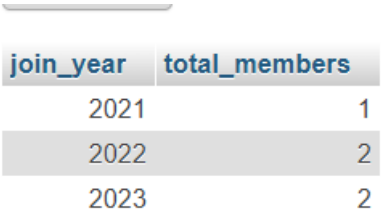
QUESTION:

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

CODE:

```
SELECT YEAR(date_of_membership) AS join_year,COUNT(*) AS total_members

FROM members

GROUP BY YEAR(date_of_membership);
```

SCREENSHOT:



join_year	total_members
2021	1
2022	2
2023	2

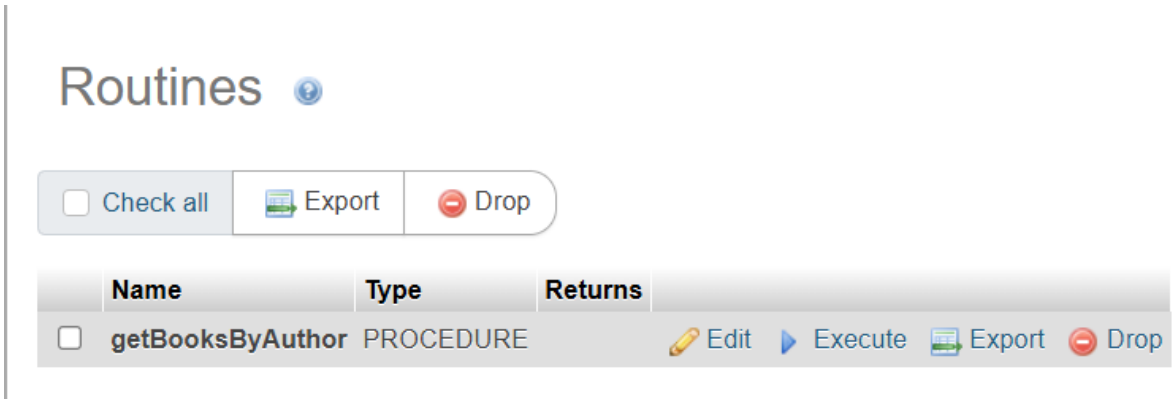
QUESTION:

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

CODE:

```
DELIMITER //

CREATE PROCEDURE getBooksByAuthor(IN authorName VARCHAR(100))

BEGIN

SELECT * FROM books WHERE author=authorName;

END //

DELIMITER ;
```

SCREENSHOT:



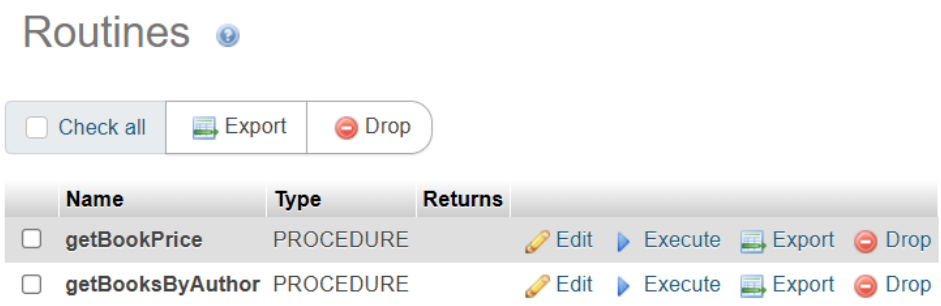
QUESTION:

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

CODE:

```
DELIMITER //

CREATE PROCEDURE getBookPrice(IN bid INT)

BEGIN

SELECT price FROM books WHERE book_id=bid;

END //

DELIMITER ;
```

SCREENSHOT:



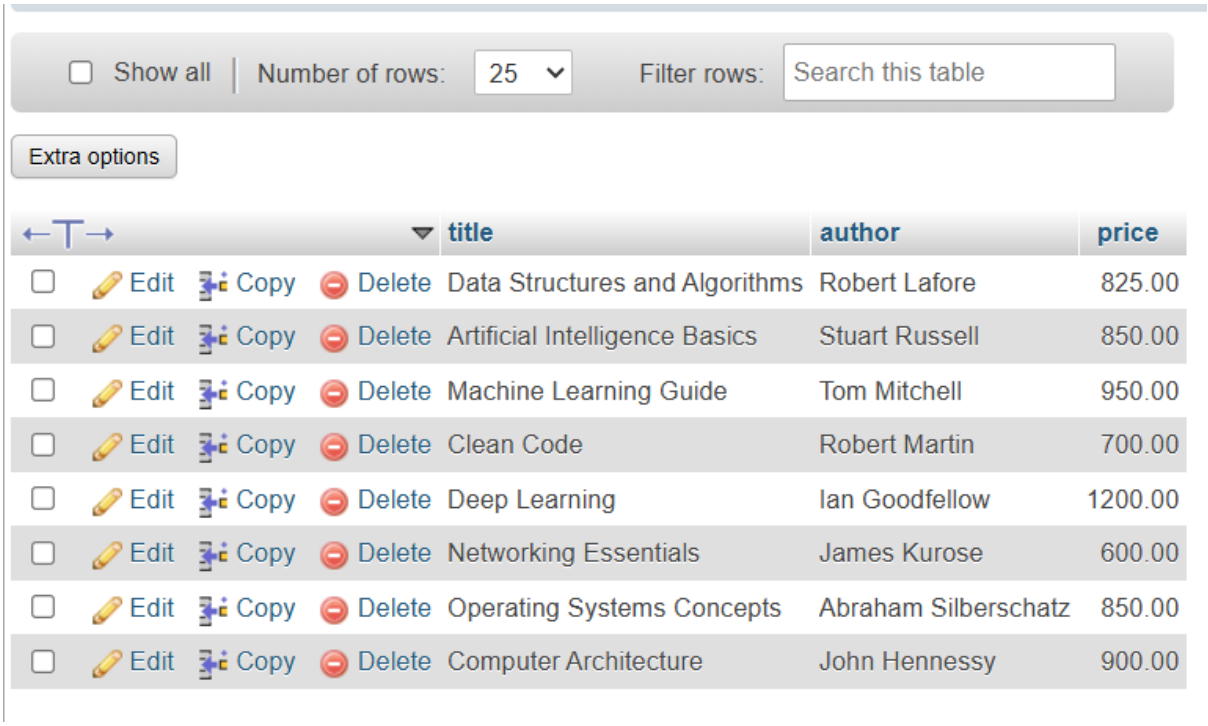
QUESTION:

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

CODE:

```
CREATE VIEW books_view AS

SELECT title,author,price

FROM books;
```

SCREENSHOT:



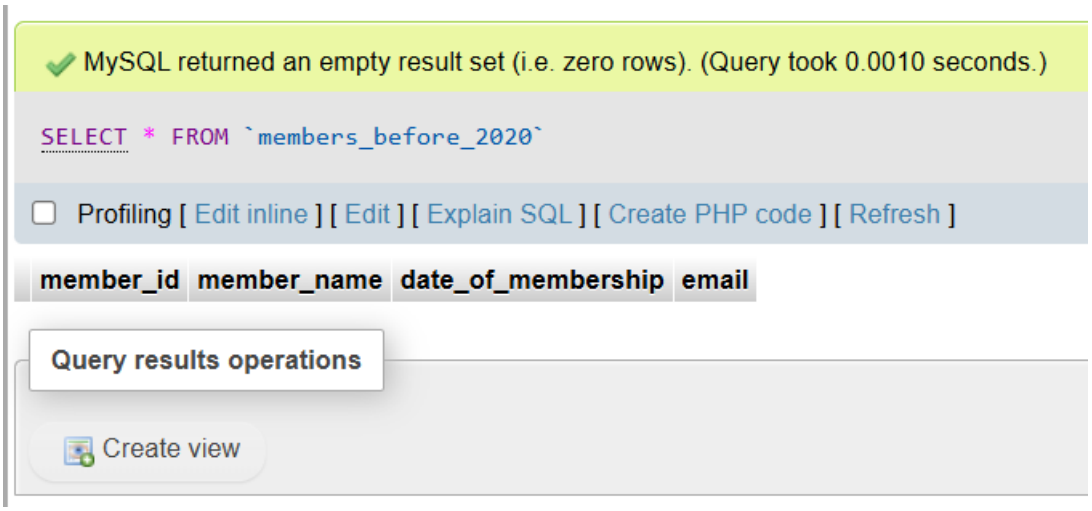
QUESTION:

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

CODE:

```
CREATE VIEW members_before_2020 AS

SELECT *

FROM members

WHERE date_of_membership<'2020-01-01';
```

SCREENSHOT:



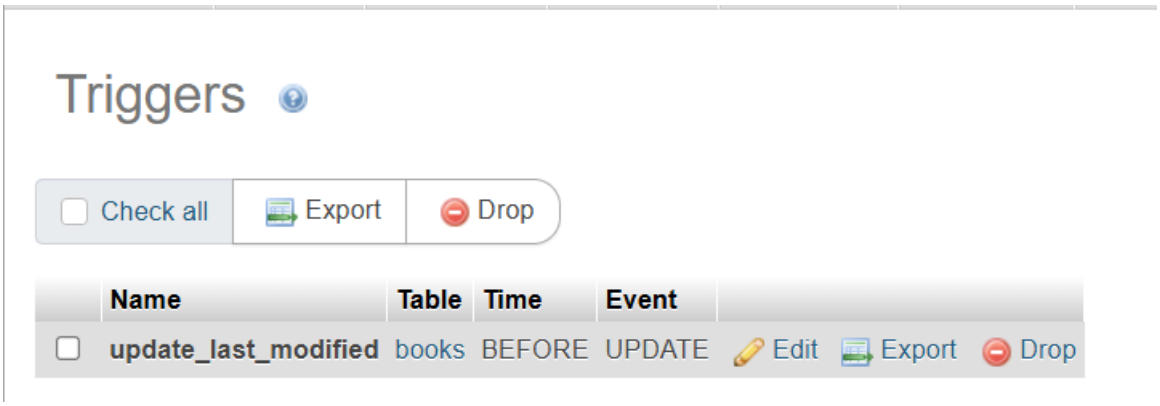
QUESTION:

Lab 3: Create a trigger to automatically update the last_modified timestamp of the books table whenever a record is updated.

CODE:

```
DELIMITER //

CREATE TRIGGER update_last_modified

BEFORE UPDATE ON books

FOR EACH ROW

BEGIN

SET NEW.last_modified=CURRENT_TIMESTAMP;

END //DELIMITER ;
```

SCREENSHOT:



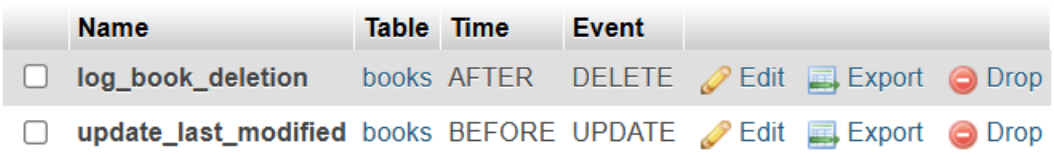
QUESTION:

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.

CODE:

```
DELIMITER //

CREATE TRIGGER log_book_deletion

AFTER DELETE ON books

FOR EACH ROW

BEGIN

INSERT INTO log_changes(book_id,title,deleted_at)

VALUES(OLD.book_id,OLD.title,CURRENT_TIMESTAMP);

END //

DELIMITER ;
```

SCREENSHOT:



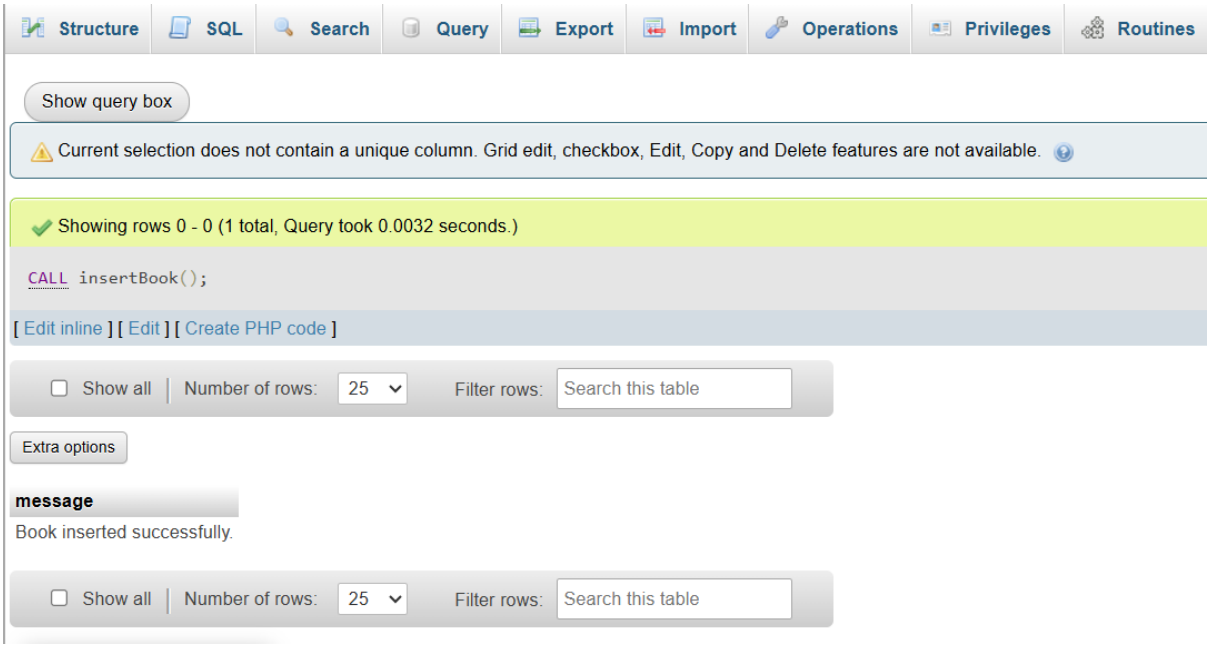
QUESTION:

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

CODE:

```
DELIMITER //

CREATE PROCEDURE insertBook()
BEGIN
    INSERT INTO books (book_id, title, author, publisher, year_of_publication, price, genre)
    VALUES (15,'Python Programming','Mark Lutz','OReilly',2023,800.00,'Programming');
    SELECT 'Book inserted successfully.' AS message;
END //

DELIMITER ;
```

SCREENSHOT:



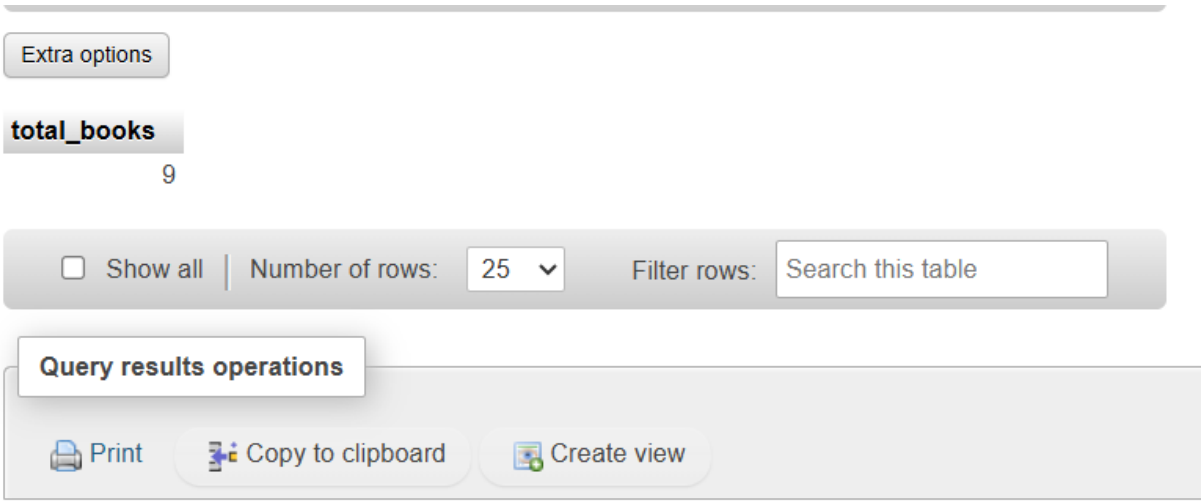
QUESTION:

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

CODE:

```
DELIMITER //


CREATE PROCEDURE getTotalBooks()

BEGIN

   SELECT COUNT(*) AS total_books FROM books;

END //


DELIMITER ;
```

SCREENSHOT:



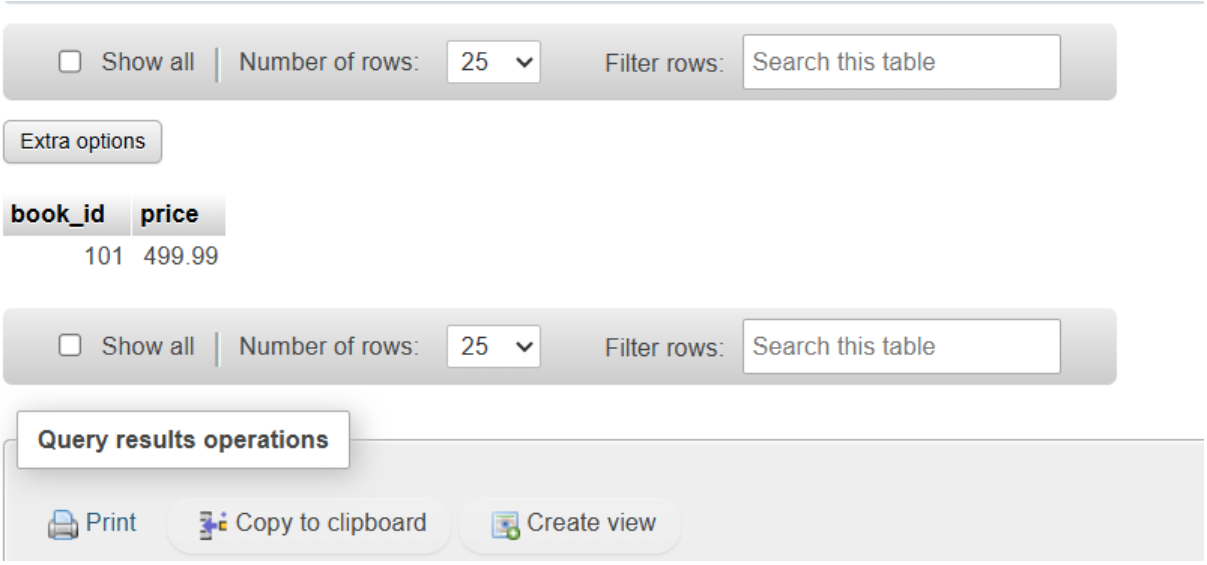
QUESTION:

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

CODE:

```
DELIMITER //

CREATE PROCEDURE showBookInfo()
BEGIN
    DECLARE v_book_id INT;
    DECLARE v_price DECIMAL(10,2);

    SET v_book_id = 101;
    SET v_price = 499.99;

    SELECT v_book_id AS book_id, v_price AS price;
END //
DELIMITER ;
```

SCREENSHOT:



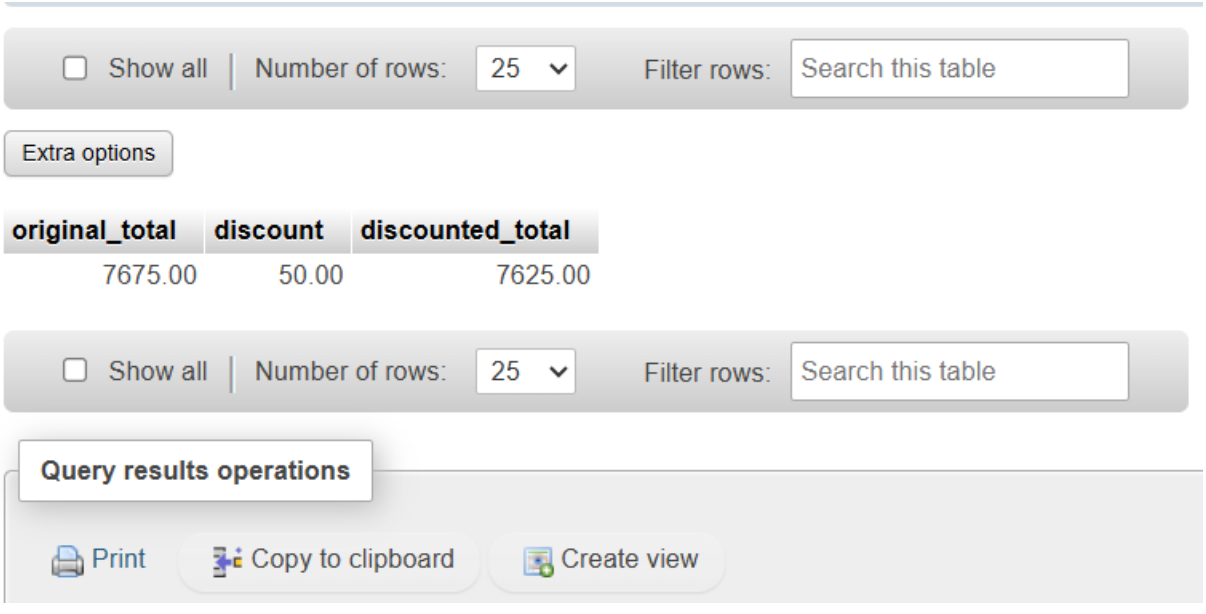
QUESTION:

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

CODE:

```
DELIMITER //


CREATE PROCEDURE priceOperations()
BEGIN
   DECLARE CONST_DISCOUNT DECIMAL(5,2) DEFAULT 50.00;
   DECLARE total_price DECIMAL(10,2);
   SELECT SUM(price) INTO total_price FROM books;
   SELECT total_price AS original_total,
        CONST_DISCOUNT AS discount,
        (total_price - CONST_DISCOUNT) AS discounted_total;
END //
DELIMITER ;
```

SCREENSHOT:



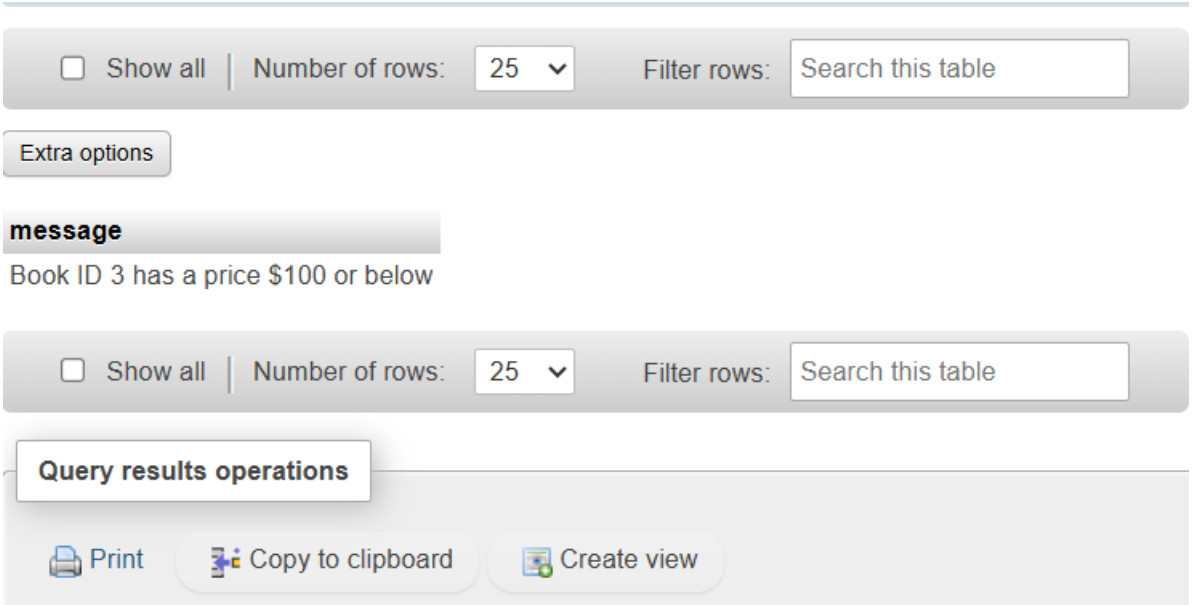
QUESTION:

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.

CODE:

```
DELIMITER //


CREATE PROCEDURE checkBookPrice(IN bid INT)
BEGIN
   DECLARE book_price DECIMAL(10,2);
  SELECT price INTO book_price FROM books WHERE book_id = bid;
   IF book_price > 100 THEN
      SELECT CONCAT('Book ID ', bid, ' has a price above $100') AS message;
   ELSE
      SELECT CONCAT('Book ID ', bid, ' has a price $100 or below') AS message;
   END IF;
END //
DELIMITER ;
```

SCREENSHOT:



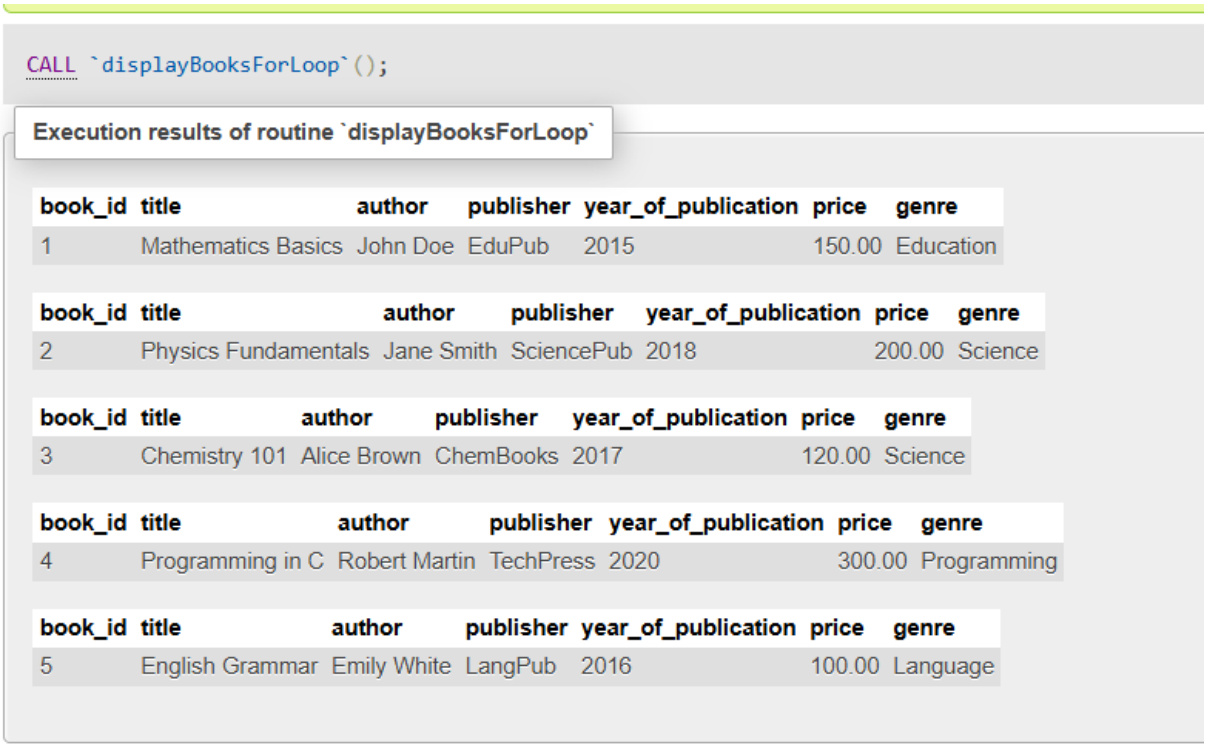
QUESTION:

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

CODE:

```
DELIMITER //
CREATE PROCEDURE displayBooksForLoop()
BEGIN
   DECLARE done INT DEFAULT 0;
   DECLARE v_book_id INT;
   DECLARE v_title VARCHAR(255);
   DECLARE v_author VARCHAR(100);
   DECLARE v_publisher VARCHAR(100);
   DECLARE v_year INT;
   DECLARE v_price DECIMAL(10,2);
   DECLARE v_genre VARCHAR(50);
 DECLARE cur_books CURSOR FOR
      SELECT book_id, title, author, publisher, year_of_publication, price, genre FROM books;
   DECLARE CONTINUE HANDLER FOR NOT FOUND SET done = 1;
 OPEN cur_books;
read_loop: LOOP
      FETCH cur_books INTO v_book_id, v_title, v_author, v_publisher, v_year, v_price,
v_genre;
      IF done THEN
     LEAVE read_loop;
      END IF;
      SELECT v_book_id AS book_id, v_title AS title, v_author AS author, v_publisher AS
publisher, v_year AS year_of_publication, v_price AS price, v_genre AS genre;
   END LOOP;
CLOSE cur_books;
END //DELIMITER ;
```

SCREENSHOT:



```
CALL `displayBooksForLoop`();
```

Execution results of routine `displayBooksForLoop`

book_id	title	author	publisher	year_of_publication	price	genre
1	Mathematics Basics	John Doe	EduPub	2015	150.00	Education

book_id	title	author	publisher	year_of_publication	price	genre
2	Physics Fundamentals	Jane Smith	SciencePub	2018	200.00	Science

book_id	title	author	publisher	year_of_publication	price	genre
3	Chemistry 101	Alice Brown	ChemBooks	2017	120.00	Science

book_id	title	author	publisher	year_of_publication	price	genre
4	Programming in C	Robert Martin	TechPress	2020	300.00	Programming

book_id	title	author	publisher	year_of_publication	price	genre
5	English Grammar	Emily White	LangPub	2016	100.00	Language

QUESTION:

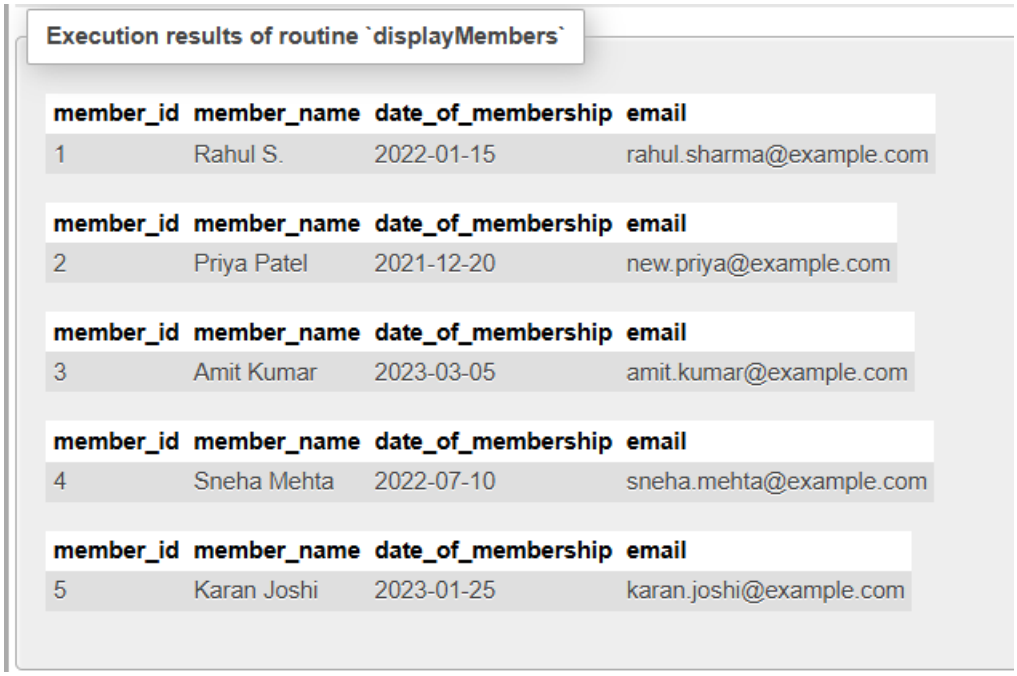
Lab 3: Write a PL/SQL block using an explicit cursor to fetch and display all records from the members table.

CODE:

```
DELIMITER //

CREATE PROCEDURE displayMembers()

BEGIN

    DECLARE done INT DEFAULT 0;

    DECLARE v_member_id INT;

    DECLARE v_member_name VARCHAR(100);

    DECLARE v_date_of_membership DATE;

    DECLARE v_email VARCHAR(100);

DECLARE cur_members CURSOR FOR

        SELECT member_id, member_name, date_of_membership, email FROM members;

    DECLARE CONTINUE HANDLER FOR NOT FOUND SET done = 1;
```

```
OPEN cur_members;

read_loop: LOOP

    FETCH cur_members INTO v_member_id, v_member_name, v_date_of_membership,
v_email;

    IF done THEN

      LEAVE read_loop;

    END IF;

    SELECT v_member_id AS member_id, v_member_name AS member_name,
v_date_of_membership AS date_of_membership, v_email AS email;

  END LOOP;

CLOSE cur_members;

END //

DELIMITER ;
```

SCREENSHOT:



Execution results of routine `displayMembers`

member_id	member_name	date_of_membership	email
1	Rahul S.	2022-01-15	rahul.sharma@example.com

member_id	member_name	date_of_membership	email
2	Priya Patel	2021-12-20	new.priya@example.com

member_id	member_name	date_of_membership	email
3	Amit Kumar	2023-03-05	amit.kumar@example.com

member_id	member_name	date_of_membership	email
4	Sneha Mehta	2022-07-10	sneha.mehta@example.com

member_id	member_name	date_of_membership	email
5	Karan Joshi	2023-01-25	karan.joshi@example.com

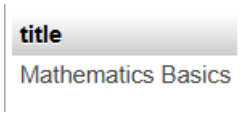
QUESTION:

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

CODE:

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

SCREENSHOT:



title
Mathematics Basics

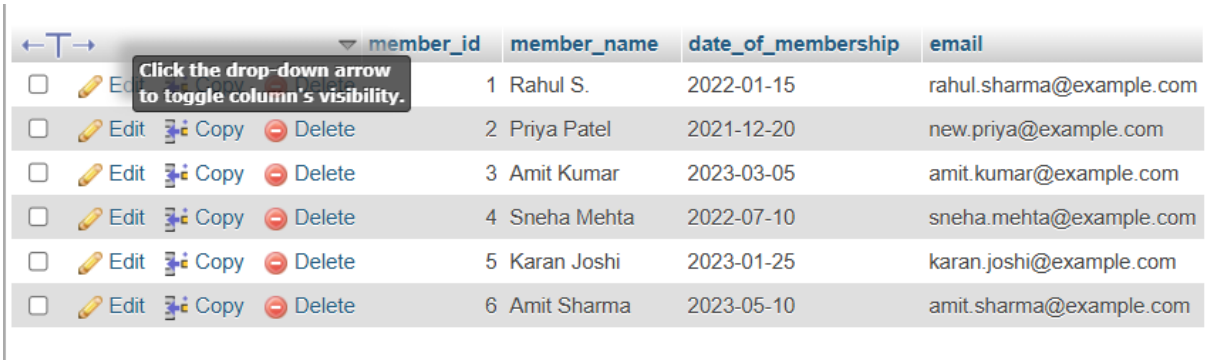
QUESTION:

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

CODE:

```
START TRANSACTION;

INSERT INTO members (member_id, member_name, date_of_membership, email)

VALUES (6, 'Amit Sharma', '2023-05-10', 'amit.sharma@example.com');

SAVEPOINT sp_before_update;

UPDATE members

SET member_name = 'Amit S.'

WHERE member_id = 6;

UPDATE members

SET email = 'amit.new@example.com'

WHERE member_id = 6;

ROLLBACK TO SAVEPOINT sp_before_update;

COMMIT;
```

SCREENSHOT:



	member_id	member_name	date_of_membership	email
Edit Copy Delete	1	Rahul S.	2022-01-15	rahul.sharma@example.com
Edit Copy Delete	2	Priya Patel	2021-12-20	new.priya@example.com
Edit Copy Delete	3	Amit Kumar	2023-03-05	amit.kumar@example.com
Edit Copy Delete	4	Sneha Mehta	2022-07-10	sneha.mehta@example.com
Edit Copy Delete	5	Karan Joshi	2023-01-25	karan.joshi@example.com
Edit Copy Delete	6	Amit Sharma	2023-05-10	amit.sharma@example.com

QUESTION:

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.

CODE:

```
START TRANSACTION;

INSERT INTO books (book_id,title,author,publisher,year_of_publication,price,genre)

VALUES

(20,'Machine Learning','John Smith','TechPress',2022,400.00,'Programming'),

(21,'Artificial Intelligence','Jane Doe','AI Pub',2021,350.00,'Programming'),

(22,'Big Data Analytics','Alice Brown','DataPub',2020,300.00,'Data Science');

COMMIT;

START TRANSACTION;

SAVEPOINT sp_before_update;

UPDATE books

SET price = price + 50

WHERE book_id IN (20,21);

ROLLBACK TO SAVEPOINT sp_before_update;

COMMIT;
```

SCREENSHOT:

☐	Edit	Copy	Delete	1	Mathematics Basics	John Doe	EduPub	2015	150.00	Education
☐	Edit	Copy	Delete	2	Physics Fundamentals	Jane Smith	SciencePub	2018	200.00	Science
☐	Edit	Copy	Delete	3	Chemistry 101	Alice Brown	ChemBooks	2017	120.00	Science
☐	Edit	Copy	Delete	4	Programming in C	Robert Martin	TechPress	2020	300.00	Programming
☐	Edit	Copy	Delete	5	English Grammar	Emily White	LangPub	2016	100.00	Language
☐	Edit	Copy	Delete	11	Data Structures	Alice Brown	TechPub	2021	250.00	Programming
☐	Edit	Copy	Delete	12	Algorithms	Bob Smith	TechPub	2020	300.00	Programming
☐	Edit	Copy	Delete	13	Database Systems	Carol White	EduPub	2019	200.00	Education
☐	Edit	Copy	Delete	14	Data Structures	Alice Brown	TechPub	2021	250.00	Programming
☐	Edit	Copy	Delete	15	Algorithms	Bob Smith	TechPub	2020	300.00	Programming
☐	Edit	Copy	Delete	16	Database Systems	Carol White	EduPub	2019	200.00	Education
☐	Edit	Copy	Delete	17	Data Structures	Alice Brown	TechPub	2021	250.00	Programming
☐	Edit	Copy	Delete	18	Algorithms	Bob Smith	TechPub	2020	300.00	Programming
☐	Edit	Copy	Delete	19	Database Systems	Carol White	EduPub	2019	200.00	Education
☐	Edit	Copy	Delete	20	Machine Learning	John Smith	TechPress	2022	400.00	Programming
☐	Edit	Copy	Delete	21	Artificial Intelligence	Jane Doe	AI Pub	2021	350.00	Programming
☐	Edit	Copy	Delete	22	Big Data Analytics	Alice Brown	DataPub	2020	300.00	Data Science