1. Створіть базу даних для керування бібліотекою книг згідно зі структурою, наведеною нижче. Використовуйте DDL-команди для створення необхідних таблиць та їх зв'язків.

a) CREATE SCHEMA LibraryManagement;

use LibraryManagement;

b) CREATE TABLE authors (

author\_id INT AUTO\_INCREMENT PRIMARY KEY,

author\_name VARCHAR(255)

);

c) CREATE TABLE genres (

genre\_id INT AUTO\_INCREMENT PRIMARY KEY,

genre\_name VARCHAR(255)

);

d) CREATE TABLE books (

book\_id INT AUTO\_INCREMENT PRIMARY KEY,

title VARCHAR(255),

publication\_year YEAR,

author\_id INT,

genre\_id INT,

FOREIGN KEY (author\_id) REFERENCES authors(author\_id),

FOREIGN KEY (genre\_id) REFERENCES genres(genre\_id)

);

e) CREATE TABLE users (

user\_id INT AUTO\_INCREMENT PRIMARY KEY,

user\_name VARCHAR(255),

email VARCHAR(255)

);

f) CREATE TABLE borrowed\_books (

borrow\_id INT AUTO\_INCREMENT PRIMARY KEY,

book\_id INT,

user\_id INT,

borrow\_date DATE,

return\_date DATE,

FOREIGN KEY (book\_id) REFERENCES books(book\_id),

FOREIGN KEY (user\_id) REFERENCES users(user\_id)

);

2. Заповніть таблиці простими видуманими тестовими даними. Достатньо одного-двох рядків у кожну таблицю.

INSERT INTO authors (author\_name)

VALUES

('Тарас Шевченко'),

('Іван Франко'),

('Леся Українка'),

('Іван Нечуй-Левицький');

INSERT INTO genres (genre\_name)

VALUES

('Detective'),

('Poetry'),

('Novel');

INSERT INTO books (title, publication\_year, author\_id, genre\_id)

VALUES

('Super story', '1874', 1, 3),

('My poetry', '1978', 2, 3),

('Very sad novel', '2002', 3, 1);

INSERT INTO users (user\_name, email)

VALUES

('Nataly Pottary', 'sfgerg@sfvs.sd'),

('Kateryna Baleryna', 'swfgdfh@sdfb.g'),

('Borys Barbarys', 'fnjhtke@dfgh.d');

INSERT INTO borrowed\_books (book\_id, user\_id, borrow\_date, return\_date)

VALUES

(3, 3, '2025-04-12', '2025-04-24'),

(2, 3, '2024-04-12', '2024-04-24'),

(1, 1, '2023-04-12', '2023-04-24');

3. Перейдіть до бази даних, з якою працювали у темі 3. Напишіть запит за допомогою операторів FROM та INNER JOIN, що об’єднує всі таблиці даних, які ми завантажили з файлів: order\_details, orders, customers, products, categories, employees, shippers, suppliers. Для цього ви маєте знайти спільні ключі.

Перевірте правильність виконання запиту.

USE hw3;

SELECT \* FROM order\_details

INNER JOIN orders ON order\_details.order\_id = [orders.id](http://orders.id)

INNER JOIN customers ON orders.customer\_id = [customers.id](http://customers.id)

INNER JOIN products ON order\_details.product\_id = [products.id](http://products.id)

INNER JOIN categories ON products.category\_id = [categories.id](http://categories.id)

INNER JOIN employees ON orders.employee\_id = employees.employee\_id

INNER JOIN shippers ON orders.shipper\_id = [shippers.id](http://shippers.id)

INNER JOIN suppliers ON products.supplier\_id = suppliers.id

4. Виконайте запити, перелічені нижче.

- Визначте, скільки рядків ви отримали (за допомогою оператора COUNT).

* USE hw3;
* SELECT COUNT(\*) FROM order\_details
* INNER JOIN orders ON order\_details.order\_id = orders.id
* INNER JOIN customers ON orders.customer\_id = customers.id
* INNER JOIN products ON order\_details.product\_id = products.id
* INNER JOIN categories ON products.category\_id = categories.id
* INNER JOIN employees ON orders.employee\_id = employees.employee\_id
* INNER JOIN shippers ON orders.shipper\_id = shippers.id
* INNER JOIN suppliers ON products.supplier\_id = suppliers.id

- Змініть декілька операторів INNER на LEFT чи RIGHT. Визначте, що відбувається з кількістю рядків. Чому? Напишіть відповідь у текстовому файлі.

USE hw3;

SELECT COUNT(\*) AS row\_count FROM order\_details

RIGHT JOIN orders ON order\_details.order\_id = [orders.id](http://orders.id)

RIGHT JOIN customers ON orders.customer\_id = [customers.id](http://customers.id)

LEFT JOIN products ON order\_details.product\_id = [products.id](http://products.id)

LEFT JOIN categories ON products.category\_id = [categories.id](http://categories.id)

LEFT JOIN employees ON orders.employee\_id = employees.employee\_id

LEFT JOIN shippers ON orders.shipper\_id = [shippers.id](http://shippers.id)

LEFT JOIN suppliers ON products.supplier\_id = suppliers.id

- На основі запита з пункта 3 виконайте наступне: оберіть тільки ті рядки, де employee\_id > 3 та ≤ 10.

USE hw3;

SELECT COUNT(\*) AS row\_count FROM order\_details

RIGHT JOIN orders ON order\_details.order\_id = [orders.id](http://orders.id)

RIGHT JOIN customers ON orders.customer\_id = [customers.id](http://customers.id)

LEFT JOIN products ON order\_details.product\_id = [products.id](http://products.id)

LEFT JOIN categories ON products.category\_id = [categories.id](http://categories.id)

LEFT JOIN employees ON orders.employee\_id = employees.employee\_id

LEFT JOIN shippers ON orders.shipper\_id = [shippers.id](http://shippers.id)

LEFT JOIN suppliers ON products.supplier\_id = [suppliers.id](http://suppliers.id)

WHERE employees.employee\_id > 3 AND employees.employee\_id <= 10;

- Згрупуйте за іменем категорії, порахуйте кількість рядків у групі, середню кількість товару (кількість товару знаходиться в order\_details.quantity)

USE hw3;

SELECT categories.name AS category\_name,

COUNT(\*) AS row\_count,

AVG(order\_details.quantity) AS average\_quantity

FROM order\_details

INNER JOIN products ON order\_details.product\_id = [products.id](http://products.id)

INNER JOIN categories ON products.category\_id = [categories.id](http://categories.id)

GROUP BY categories.name;

- Відфільтруйте рядки, де середня кількість товару більша за 21.

USE hw3;

SELECT categories.name AS category\_name,

COUNT(\*) AS row\_count,

AVG(order\_details.quantity) AS average\_quantity

FROM order\_details

INNER JOIN products ON order\_details.product\_id = [products.id](http://products.id)

INNER JOIN categories ON products.category\_id = [categories.id](http://categories.id)

GROUP BY [categories.name](http://categories.name)

HAVING AVG(order\_details.quantity) > 21;

- Відсортуйте рядки за спаданням кількості рядків.

USE hw3;

SELECT categories.name AS category\_name,

COUNT(\*) AS row\_count,

AVG(order\_details.quantity) AS average\_quantity

FROM order\_details

INNER JOIN products ON order\_details.product\_id = [products.id](http://products.id)

INNER JOIN categories ON products.category\_id = [categories.id](http://categories.id)

GROUP BY [categories.name](http://categories.name)

HAVING AVG(order\_details.quantity) > 21

ORDER BY row\_count DESC;

- Виведіть на екран (оберіть) чотири рядки з пропущеним першим рядком.

USE hw3;

SELECT categories.name AS category\_name,

COUNT(\*) AS row\_count,

AVG(order\_details.quantity) AS average\_quantity

FROM order\_details

INNER JOIN products ON order\_details.product\_id = [products.id](http://products.id)

INNER JOIN categories ON products.category\_id = [categories.id](http://categories.id)

GROUP BY [categories.name](http://categories.name)

HAVING AVG(order\_details.quantity) > 21

ORDER BY row\_count DESC

LIMIT 4 OFFSET 1;