Task 7: Creating Views in SQL

# Objective:

Learn to create and use views.

# Tools:

DB Browser for SQLite / MySQL Workbench

# Outcome:

Understand reusable SQL logic through abstraction and security.

# Library Database Schema (Assumed):

-- Book Table  
CREATE TABLE Book (  
 BookID INT PRIMARY KEY,  
 Title VARCHAR(100),  
 Author VARCHAR(100),  
 CategoryID INT,  
 PublisherID INT  
);  
  
-- Borrower Table  
CREATE TABLE Borrower (  
 BorrowerID INT PRIMARY KEY,  
 Name VARCHAR(100),  
 Email VARCHAR(100)  
);  
  
-- Issue Table  
CREATE TABLE Issue (  
 IssueID INT PRIMARY KEY,  
 BookID INT,  
 BorrowerID INT,  
 IssueDate DATE,  
 ReturnDate DATE  
);

# 1. Creating Views

## Example 1: View to Show Currently Issued Books

CREATE VIEW CurrentIssuedBooks AS  
SELECT   
 b.Title,  
 br.Name AS BorrowerName,  
 i.IssueDate,  
 i.ReturnDate  
FROM   
 Issue i  
JOIN Book b ON i.BookID = b.BookID  
JOIN Borrower br ON i.BorrowerID = br.BorrowerID  
WHERE i.ReturnDate IS NULL;

Purpose: Lists books that are currently issued (not yet returned).

## Example 2: View to Show Book Details with Author and Publisher

CREATE VIEW BookDetailsView AS  
SELECT   
 b.BookID,  
 b.Title,  
 b.Author,  
 c.CategoryName,  
 p.PublisherName  
FROM   
 Book b  
JOIN Category c ON b.CategoryID = c.CategoryID  
JOIN Publisher p ON b.PublisherID = p.PublisherID;

Purpose: Abstracts complex joins and shows full book info in one simple view.

# 2. Using Views

-- Get all currently issued books  
SELECT \* FROM CurrentIssuedBooks;  
  
-- Get book details where publisher is 'Penguin'  
SELECT \* FROM BookDetailsView  
WHERE PublisherName = 'Penguin';

# 3. Benefits of Views

* Abstraction: Hides complex JOINs and filters
* Reusability: Can be reused in multiple queries
* Security: Exposes only selected columns and rows
* Maintainability: Centralized logic, easier to update

# 4. Dropping a View

DROP VIEW IF EXISTS CurrentIssuedBooks;

# Deliverables (Summary):

* CREATE VIEW examples with JOINs and filters
* SELECT usage from views
* DROP VIEW example