

Phase 2: Building a Relational Database Management System

Due: March 24, at the beginning of class

What to Turn In:

- Submit an electronic copy of Phase 2 of your project by "***submit cchen 3100-phase2 your_files***"
Your submission should include:
 - a README file of how to install and run your software, and names of the team members
 - all of your HTML, PHP and SQL code for Phase 2
- You should pre-record a 5-7 minutes video demo and present it during class on March 24 to demonstrate how your project works.
(one submission/demo per team)

NOTE: Do not use client-side script such as JavaScript, CSS, etc. Only use plain HTML and PHP.

During Phase 1, you have designed the E-R diagram and relational schema of the database. In Phase 2, you will enforce the constraints and implement the desired functionalities as specified in Phase 1's project description.

Your application should have the following functionalities:

1. There is a super user who can modify all contents of the database.
2. Publishers can add new books.
3. Customers can register, and login with username and password, and update their own information. Customers may become members after paying membership fee.
4. Customers may make purchases of books. Customers should be able to check their order information and purchase history.
5. Customers should be able to search by keyword.
6. Different types of books may have different delivery method.
7. Customers may give books they have purchased a rating between 1 and 5 and a comment.

Phase 2 of the project contains two parts.

Part 1: Create and populate the database:

1. Create a database called "bookstore" for user "root". Do not specify password for root.
2. In "bookstore", create tables by running a "DB2.sql" file and populate the tables with your own fabricated data.
 - Specify primary key for each table.
 - Tables that are translated from relationship sets, weak entity sets, and subclasses should have foreign key constraints specified.
 - Each table should contain at least 5 records. Do not insert more than 50 records for any table.
 - Put all "CREATE TABLE" and "INSERT INTO" statements in one DB2.sql file so your database can be easily duplicated.

Part 2: Perform the following tasks as queries (some queries may include several steps):

1. A new customer registers, upgrades from non-member to member.
2. A publisher adds a new book with author information to the database, updates price of a book.
3. The admin (super user) updates the cost of shipping methods for books.
4. A customer searches for a particular book by title and/or author and purchases the book.
5. A guest searches for the best-selling book of a given year, if no year is given, return the best-selling book for the entire history.
6. A customer checks their order history and reorder a book
7. An author purchases their own books.
8. A customer gives rating and comment to a book they have purchased, checks rating and comments of a book.

You should have one *.html* file as user-interface that takes user input. You may have one short *.php* file for every query or one long *.php* file for all the queries combined.