Library Management System

The Faculty College of Computer Studies

University of Antique

Sibalom, Antique

In Partial Fulfillment of the

Requirements of Database System 1

Presented by

Nerissa Mae S. Coloma

Ma. Dorothy E. Cumla

Aikrylle C. Gilua

Leevan Dave Osorio

Marlowe MiJoy E. Ovivir

Vannessa Jean Santerva

Renalyn Y. Villaflor

BS INFO 2B

This is the **home page of a local web-based Library Management System** called **CCS Library System**, running on localhost. It features a navigation bar with links for managing users, students, books, borrowing/returning, and more. The main content area welcomes users with placeholder text, and a footer shows copyright info for 2024.

A screenshot of a computer

Description automatically generated

This section allows administrators to **view, add, edit, or delete user accounts** within the library system. It provides a table displaying essential details for each user.

A screenshot of a computer

Description automatically generated

This page is the **Student Management section** of the **CCS Library System**. It allows administrators to manage student records by viewing, adding, editing, or deleting entries. The page displays a table listing each student's ID, name, address, and course, along with action buttons for editing or deleting each record. A "+" button is available for adding new students.

A screenshot of a computer

Description automatically generated

This page presents a tabular view of various academic courses, each with an id, code, course name and department. For each entry, there are edit and delete action buttons, allowing for modification or removal of the course information. Above the table, an add new course, indicating functionality to add new course entries to the system.

A screenshot of a computer

Description automatically generated

This section provides a clear, tabular overview of books that have been borrowed. At the top of the page, there is a prominent borrow books button, which would likely be used to initiate the process of lending out a new book. Below this, a table lists individual borrowing records, each detailing the userid, username studentid, title of the book date borrowed, and date returned. For each record, there are edit and delete action buttons, enabling library staff to modify or remove existing borrowing entries. Overall, the interface provides a functional way to track and manage book lending activities within the library system.

A screenshot of a computer

Description automatically generated

This page is designed for maintaining a database of authors. It features an add author button, which presumably allows administrators to input details for a new author into the system. Below this button, a table lists existing author records, each containing an id, first name, middlename and last name. Each row also includes edit and delete action buttons, enabling the modification or removal of author information.

A screenshot of a computer

Description automatically generated

Specifically showing the operations performed on the student table within the clsdb database. The main section of the screen shows a series of successful insert queries that have been executed. Each green highlighted box indicates that "1 row inserted" and provides details about the query execution time.

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

Particularly showing the contents of the authors table within the clsdb database. The main section of the screen presents the results of a select query: select id, first name, middle name, last name from authors which has successfully retrieved 9 rows

A screenshot of a computer

Description automatically generated

The same applies to the student.

A screenshot of a computer

Description automatically generated

Specifically showing the results of a SQL query executed on the course table within the clsdb database. The query, select department count (\*) as total courses from course group by department with rollup, aims to count the number of courses per department and provide a grand total.

A screenshot of a computer

Description automatically generated

Showing the execution of a SQL query on the course table within the clsdb database. The SQL query itself is a union operation, combining two select statements. The first select statement retrieves coursed, code, course name, and department from the course table where the department is CMS. The second select statement, after the union keyword, similarly retrieves the same columns from the course table but for records where the department is CEA.

A screenshot of a computer

Description automatically generated

Shows the initial state, where set autocommit = 0 and begin, the statements have been executed. These commands are typically used to start a database transaction, allowing multiple SQL statements to be treated as a single, atomic operation, which can either be fully committed or entirely rolled back. Both operations returned empty result set, which is expected as they are command executions, not data retrievals.

A screenshot of a computer

Description automatically generated

Shows an insert statement and an update statement being prepared in the SQL query box. The insert statement insert into books aims to add a new record to the books table. Following this, an update statement, update is prepared to modify the title of the book with bookid 1.

A screenshot of a computer

Description automatically generated

Displays the results of executing these prepared SQL statements within the transaction. It confirms that 1 row inserted because of the insert statement, and subsequently, 1 row affected by the update statement. Finally, a commit, statement is executed, which permanently saves all the changes made within the transaction to the database. This final operation also returns an empty result set, as it's a transactional command. The series of actions demonstrates a common workflow in database management starting a transaction, performing data manipulation (insertion and update), and then committing the changes.

A screenshot of a computer

Description automatically generated