## 

## 

**Experiment 5:**

**Data Modeling and Database**

CPE106L (Software Design Laboratory)

**Member 1: Claros, Angelica A.**

**Member 2: Facal, Ma. Catherina N.**

**Member 3: Santos, Angelica Anne D.**

Group No.: **5**

Section: **B2**

## **PreLab**



|  |
| --- |
| **Laboratory Insights** |

**Claros**

In this final laboratory, the focus was on data and the importance of proper storage and collection methods. Just like research requires complete and organized data, software programs also require data to be processed effectively. In previous labs, I learned how to build programs, and in this lab, I gained insight into visualizing and upgrading programs to store, save, and delete data efficiently.

From our discussion, I realized that a good database is crucial for a program. I learned about MySQL and DB Browser as suitable tools for managing databases. While MySQL is commonly used for managing and manipulating databases, DB Browser is mainly used for creating, designing, and editing files in databases. In conclusion, I understood that a user interface alone is insufficient for an efficient application; effective database management is equally important. By mastering database management and software application development skills, we can enhance application efficiency and contribute to projects that benefit the common good.

**Facal**

The fifth laboratory session emphasizes the significance of databases in managing and organizing data effectively. The main takeaway for me is understanding Entity-Relationship Diagrams (ERDs), which serve as visual aids for illustrating relationships between entities within a database. In simpler terms, ERDs depict how various pieces of information are interconnected, offering insights into the data structure.

Effective database design plays a pivotal role in streamlining operations, eliminating redundancy, optimizing data access, and expediting data manipulation processes. This, in turn, enhances program efficiency and overall performance. By mastering concepts such as ERDs and normalization, organizations can harness the power of databases to maximize the utility of their data assets. In summary, databases are integral components of modern computing, offering indispensable support for data-driven decision-making. A thorough understanding of database design and administration principles is crucial for businesses to thrive in today's digital landscape.

**Santos**

The fifth laboratory exercise underscores the importance of databases and how they facilitate data management and organization. Essential to this learning is the concept of Entity-Relationship Diagrams (ERDs) – which are visual tools used to represent the relationships between entities in a database. In simpler terms, they show how different pieces of information are connected. By understanding ERDs and database design, businesses can improve operations and outcomes.

Databases come in various forms and serve diverse purposes depending on the application's goals. They provide consistency, accuracy, and security for effective data administration. In this exercise, SQLite was introduced as a tool for organizing various data types with Python. MySQL, for instance, is a powerful tool for creating and managing databases. Applications like Visual Studio Code with Python extensions can facilitate data retrieval and connection to programs. Moreover, smart database design eliminates redundancy, optimizes data access, and speeds up data manipulation, ultimately enhancing program efficiency.

By mastering concepts like ERDs, normalization, and database design, organizations can leverage databases to make more efficient use of their data. Overall, databases play a pivotal role in modern computing, and understanding their design and administration principles is essential for businesses to thrive in the digital era.

**Questions and Answers**

1. DML and DDL in SQL: DML (Data Manipulation Language) statements are used for manipulating data in the database, including INSERT, UPDATE, and DELETE. DDL (Data Definition Language) statements are used for defining or altering the structure of the database, such as CREATE TABLE, ALTER TABLE, and DROP TABLE.
2. SQLite functions are grouped into Scalar Functions (e.g., LENGTH, UPPER, LOWER), Aggregate Functions (e.g., SUM, AVG, COUNT), and Date and Time Functions (e.g., DATE, TIME, STRFTIME).
3. To check if SQLite is installed in the system using the Linux terminal, the command sqlite3 --version can be used. This command displays the installed SQLite version, if available, or returns an error message if SQLite is not installed.