[CS-499-19912-M01 Computer Science Capstone 2025](https://learn.snhu.edu/d2l/home/2019781)

5-2 Milestone Four: Enhancement Three: Databases

Eddy Kwon

10/05/2025

For my final enhancement in the database category, I chose to extend my *Advising Assistance Program* from CS 300, which I had previously enhanced for software design and algorithms. The original version of the program was designed to load course data from a CSV file, store it in memory, and allow users to search, display, and sort courses. While this worked well for small data sets, it lacked scalability, persistence, and the reliability of a true database system. For this milestone, I focused on integrating a relational database component to replace the CSV-based structure. I used SQLite to manage the course information, including course numbers, titles, and prerequisite relationships, and created SQL queries that allowed the program to load and retrieve course data dynamically from the database. This approach not only modernized the artifact but also demonstrated my understanding of real-world data storage and retrieval systems.

I selected this artifact for enhancement because it allowed me to bridge my previous work in C++ data structures with practical database implementation skills. By moving from a flat file to a relational model, I was able to showcase key database concepts such as normalization, indexing, and query optimization. The enhanced system ensures data integrity and reduces redundancy by storing courses and prerequisites in separate tables connected by foreign keys. In addition, I implemented SQL error handling and validation within the program to prevent crashes caused by missing or duplicate entries. This enhancement directly aligns with the Computer Science program outcomes related to software engineering and database design, as it demonstrates my ability to apply innovative and well-founded techniques to produce efficient and reliable computing solutions.

The enhancement process taught me the importance of balancing performance with maintainability. One of the challenges I encountered was designing an efficient schema that could handle many-to-many relationships between courses and prerequisites without complicating the query logic. I solved this by creating a junction table that mapped course relationships and by testing multiple query structures to ensure optimal performance. I also learned how to parameterize SQL queries within C++ to prevent SQL injection vulnerabilities, improving the overall security of the program. These improvements gave me a much deeper understanding of how databases interact with application-level code and how to design secure and scalable systems for real users.

Through this enhancement, I met my planned outcomes for database integration and demonstrated substantial progress toward the professional competencies expected in the CS499 capstone. I now have an artifact that connects all areas of my learning—from software design to algorithm efficiency and now to database architecture—showing a complete and well-rounded system that reflects my growth as a computer science professional. The experience reinforced how crucial data persistence, normalization, and security are for the success of any software solution. It also helped me gain confidence in working with SQL and C++ integration, skills that will be directly valuable in my future career as a software engineer.