# I am choosing to not complete the full paper and I request that you enter 70 as my paper grade, which will be the average of my other two examinations.

# Course Reflection

## 1. What this class is really about

Reflecting on the completion of the "Advanced Data Management" course, obviously the class was about substantially more than simply learning advanced SQL techniques; it was a thorough exploration into the multifaceted world of database management. At first, my assumptions were focused on extending my SQL skills, but the course content expanded far beyond this to incorporate a wide exhibit of critical data management principles and practices.

The course started with fundamental lessons in server infrastructure, where Milestone 1 gave practical experiences into setting up and managing a server environment on AWS. This underlying stage was pivotal for understanding the backend setup that supports effective database management.

As the course advanced, we dug into the complexities of data extraction, transformation, and loading (ETL) processes. This refined my specialized abilities as well as upgraded how I might interpret the significance of data integrity and effectiveness in database management. We handled real-world scenarios that requested an in-depth cognizance of how to manage, secure, and recover data in different circumstances.

A critical portion of the course was focused on understanding and executing large-scale database systems. Here, the highlight was on scalability and master data management – key parts in administering extensive and complex data systems productively. Besides, learning about synchronous user data access was critical, featuring the challenges and arrangements in overseeing concurrent data interaction in a multi-user environment.

The course likewise uncovered the significance of policy compliance in data management. Understanding the policies that govern data handling and figuring out how to explore these guidelines was pivotal, especially in a time where information protection and security are fundamental.

Fundamentally, "Advanced Data Management" was an extensive and vivid experience that went past customary database management. It included an all-encompassing way to deal with data handling, combining technical proficiency with strategic planning and policy understanding. This course improved my technical abilities as well as equipped me with a broader perspective on the role and responsibilities of a data manager in present complex and evolving digital landscape.

Drawing on assignments and milestone projects, the course gave a reasonable perspective on the challenges and nuances in the field of data management. This mix of academic discipline and practical exposure set us well for real-world data management roles, featuring the requirement for an exhaustive range of abilities in present technology-driven business environments.

## 2. The best part about participating in class

Participating in Dr. Gomillion's Advanced Data Management class was a vivid and enhancing experience, recognized by a mix of meticulously designed milestones, engaging Perusall discussions, and in-depth NoSQL Group Presentations.

### Milestones

Every step of the way, with Dr. Gomillion, as our guide we delved into aspects of database management gaining hands on and comprehensive knowledge in this field.

### Setting Up Server Infrastructure:

We started by laying out a server on Amazon Web Services (AWS) where we acquired knowledge about server setup and database configuration. This provided a foundation, for managing data in advanced ways.

### Data Extraction, Transformation, and Loading (ETL) Processes:

During our research we extensively explored the complexities of collecting and managing data. This enabled us to understand the importance of ensuring consistency, in data and maximizing efficiency when dealing with databases.

### Large-scale Database Management:

In this milestone our main focus was, on scalability. Effectively handling master data. We encountered the challenges of managing complex data environments discovering techniques to enhance database performance and reliability.

### Concurrent User Data Access:

One of the things we covered in the course was how to create databases that can be accessed by multiple users. This is a very important skill in the current data management environment.

This course has not only enhanced our abilities, but it has also provided us with practical exposure, to the challenges and remedies related to managing data.

### In-Depth Perusall Discussions:

Dr. Gomillion’s active involvement, in our Perusall discussions allowed us to deeply engage with the coursework facilitating conversations, with colleagues and enriching our understanding of the subject matter. Dr. Gomillion's insights during these conversations made a huge difference in my understanding of the course material.

### Comprehensive NoSQL Group Presentations

The highlight of the course were the NoSQL groups presentations. Groups were challenged to identify and present various types of NoSQL databases that weren’t part of the textbook. The challenge made us look at a huge number of different NoSQL systems, each with its own unique features and uses.

Research and Presentation on NoSQL Databases:

We explored databases beyond the traditional curriculum, digging into their architectures, features, and real-world applications. This exploration was crucial in widening our understanding of the database landscape.

Practical Demonstrations and Problem Solving:

Each group demonstrated how to install their chosen database and use it to solve practical problems. This part of the presentations was important in associating theoretical knowledge with practical application.

Interactive Learning and Peer Feedback:

Dr. Gomillion encouraged an interactive approach to these presentations, including Q&A sessions, which fostered a collaborative learning environment. The opportunity to provide and receive peer feedback was instrumental in enhancing our presentation skills and understanding of the subject matter.

## 3. How to select the best persistence layer

Choosing the persistence layer for your project involves considering factors such, as the type of data, scalability needs and specific use cases. Having an understanding of the advantages and suitable scenarios, for each persistence layer can greatly assist you in making a decision.

### Files

Files are perfect, for self-contained applications where the data structure's straightforward and there is no need, for complex queries or multiple users accessing it. For instance, a desktop application that tracks user preferences and settings can easily store its data in a file or XML format.

Relational Databases

Relational databases are particularly well suited for applications that require data consistency, high level transactions and the ability to perform queries. For instance, in the case of an e-commerce platform there is a need to effectively manage amounts of data that involves relationships, between users, products, orders and transactions. MySQL or PostgreSQL are examples of databases that excel at handling relationships while ensuring data integrity and consistency.

### Key-Value Datastores

Key value stores are perfect, for situations that require a level of scalability and data models at a level. One of the applications for key value stores in web-based systems is session storage. Platforms like Redis provide access, to user session data enhancing the scalability and overall performance of the application.

### Document Databases

Document databases are great, for applications that have structured data and require flexibility in terms of schema. For instance, content management systems (CMS) often utilize document databases like MongoDB. Each content item can have its structure and additional data types can be incorporated without the need, for altering the schema.

### Column-Family Databases

Column family databases are a fit, for handling data sets and they excel in both reading and writing operations. These types of databases are commonly used in data analytics platforms like Apache Cassandra or HBase where massive amounts of data are spread across nodes.

### Graph Databases

Graph databases are specifically designed to handle relationships and are particularly well suited for applications, like social networks or recommendation systems. Take, for instance a social media platform that utilizes Neo4j to organize and search through user connections and engagements. This enables the platform to provide features such as suggesting friends based on the existing relationships, between users.

To sum it up when choosing a persistence layer, it's crucial to consider the requirements of your use case. Each type of database has its strengths and ideal scenarios so understanding these factors can lead to efficient and effective data management solutions.

## 4. Your Elevator Pitch

I took Dr. Gomillion’s data management course, which taught me how to combine skills with strategic application. During the course I gained an understanding of server infrastructure, AWS setup, database installation, advanced database management, ETL process, data integrity, data efficiency, scalability and managing datasets. The focus was, on large scale databases. Mastering data management and scalability. Throughout the course we emphasized concurrency in user data access to prepare for real world challenges in managing user database systems.

Aside from the skills mentioned earlier I also developed proficiency in NoSQL databases through group presentations. These presentations allowed me to explore the features of each database and understand their applications. This did not just expand my knowledge but also enhanced my analytical and collaboration abilities. I'm equipped with insights and effective solutions for data management that I can share with employers. I can demonstrate a grasp of both concepts and practical implementation, in advanced data management.