Mihir Patel

CS 499

Prof. Sherin Aly

August 10, 2025

This artifact is a SQL-based database solution that consists of structured table creation, data transformation logic, and a final analytic view. The database includes two main tables: raw\_customer\_data (to store incoming raw churn data) and churn\_features (which contains engineered features derived from raw attributes). Finally, a view named churn\_analysis\_view is created to consolidate data for analytical consumption.

I chose this artifact because it highlights my ability to work with databases—creating schema, transforming raw data, implementing business logic using SQL, and preparing data for downstream analysis. The artifact showcases my skills in:

* Designing normalized and structured tables for analytical use cases.
* Performing inline feature engineering using SQL (e.g., churn risk score).
* Handling edge cases like zero-tenure division using CASE statements.
* Creating readable and reusable views for reporting or visualization layers.
* Implementing secure and role-based access control using SQL GRANT statements.

This artifact supports industry-standard practices in database design and showcases a key competency in transforming raw transactional data into meaningful insights.

Refactored the SQL script to eliminate hard-coded values with better control structures. Added a GRANT statement for controlled access to schemas and tables, following best security practices. Introduced robust CASE logic to handle potential issues like division by zero when calculating average\_monthly\_charge. Modularized the logic by creating a separate churn\_features table instead of applying transformations directly within the view. Designed a reusable churn\_analysis\_view to centralize key attributes from both raw and feature tables, making it easier to connect to Power BI or other analytics tools.

Demonstrate an ability to use well-founded and innovative techniques, skills, and tools in computing practices for the purpose of implementing computer solutions that deliver value and accomplish industry-specific goals (software engineering/design/database).

Working on this SQL artifact helped me solidify my understanding of relational data modeling and feature transformation in SQL. I learned the importance of writing clean and efficient SQL scripts that are not only functional but also maintainable and scalable. One of the challenges was designing logic that gracefully handled edge cases, such as tenure = 0, which could have caused division errors.

Another learning moment was building the final view in a way that can easily support reporting dashboards like Power BI, without having to repeat transformation logic. I also appreciated the opportunity to practice security management with GRANT permissions and to think from a data governance perspective.

Overall, this enhancement has helped me develop confidence in designing data pipelines in SQL and producing analytics-ready data in a secure and professional way.