Unit 6: Database Design and Normalisation

Overview

In this unit, the focus was on understanding how databases are structured, how data is cleaned for storage and use, and how **normalisation** helps ensure reliability and efficiency. We explored the foundations of relational databases—including the use of **primary and foreign keys**—and how to design clean, scalable systems that minimise redundancy and anomalies.

Key Learning Areas

- **Data Cleaning & Storage**
  We looked at how various cleaning methods support better data organisation, helping ensure the data stored in a database is consistent, accessible, and easy to manage.
- **Relational Database Structure**
  I gained a clearer understanding of how databases are built**;** particularly how key fields (primary and foreign keys) are used to link tables and maintain data integrity across the system. This built on knowledge I had already gained through prior self-study in data analysis.
- **Anomalies and Data Integrity**
  We explored how anomalies—such as duplication, insertion, and deletion issues—can compromise the accuracy of a database if not addressed during the design and cleaning process.
- **Normalisation and Its Importance**
  Normalisation was a core topic in this unit. I learned how applying different **normal forms** improves database efficiency by reducing redundancy and preventing anomalies. It's a key part of ensuring that the data remains consistent as the system scales or changes.

Team Project: Database Design

As part of a group assignment, I worked with peers to design and build a **logical database** in the role of software consultants and developers. Our project included:

- **Planning the structure** – Defining key data entities, their attributes, how they relate, and which formats and data types best represent them.
- **Creating a database design proposal** – Outlining the tools and systems (e.g. PostgreSQL) chosen based on client requirements for data storage, access, and performance.
- **Reviewing the data pipeline** – Explaining how data was captured and cleaned, including the step-by-step techniques we used to prepare it for integration into the database.

You can view our full project here:

https://github.com/TechieMaks/eportfolio.github.io/blob/eb47943daba6d2e9114b67ca1459d81f1e2e0a84/Team%20Project%20Final.pdf

Personal Reflection

This unit brought together everything we've been learning—data quality, structure, and real-world application. Collaborating on the team project gave me practical experience in translating theory into a working database design.

That said, working with my peers had its challenges. Some contributions didn't quite meet the standard I expected, and it reminded me that teamwork often involves navigating different levels of commitment, understanding, and approach. But that's the reality of collaboration—**not everyone will think or act like you do**, and that's part of the learning curve.

As an entrepreneur who's used to doing things a certain way, this experience also reminded me of working with external vendors or freelancers—where delays or misalignment can happen despite your best planning. It reinforced the importance of clear communication, patience, and building systems that allow flexibility while keeping the end goal in focus.

I now have a better appreciation for how structured design and clean data contribute to building scalable, reliable systems—especially when working with others in a real development setting.

Required Reading

- McKinney, W. (2022) Python for Data Analysis: Data Wrangling with Pandas, NumPy, and Jupyter. 3rd edn. Sebastopol, California: O'Reilly.
    - Chapter 13.