**Planning**

**Course Materials:**

* The Relational Data Model
  + FD & Keys
  + BCNF Normalisation
  + Referential Integrity
* Complex SQL Queries
  + Grouping & Aggregation
  + SQL DML

**Sprint 2 goals:**

* accurately determine and depict the multiplicities of binary and multi-way relationships in an entity-relationship diagram (ERD) and describe how the role of identifiers in a relationship depends on those multiplicities. (Multiplicity)
* identify plausible functional dependencies for a set of attributes and use them to determine the set of keys and super keys for a relation. (FD & Keys)
* prevent data anomalies among a set of relations by decomposing a relation into BCNF using its FD’s. (BCNF Normalisation)
* document referential integrity constraints in relational algebra and implement foreign keys with desired referential integrity constraint policies using SQL DDL. (Referential integrity)
* construct SQL queries using the GROUP BY and HAVING clauses to capture OLAP-style aggregation logic. (Grouping & Aggregation)
* construct SQL DML queries to declaratively modify the contents of tables. (SQL)

**How to measure at the end of the sprint:**

* Identify the relationships between entities.
* Identify the set of keys and super keys for a relation.
* Eliminate data anomalies using BCNF.
* implement foreign keys using SQL DDL.
* Write some advanced SQL code.

**Course Level Competency**

* Data Analytics
  + level 1: all (evaluated by SQL codes and database output)
  + level 2:
    - Expresses complex logic as single SQL queries using aggregation and sub-queries (evaluated by SQL query examples)
    - Understands how functional dependencies and referential integrity affect the semantics of queries (evaluated by SQL query examples)
    - Describes the logical ordering of operators in complex queries that involve nested logic (depends on course scope and progress, will consult TA)
* Data Modelling
  + level 1:
    - Appends newly acquired data to pre-existing tables, modifying their structure as necessary.
  + level 2:
    - Eliminates data anomalies with effective normalisation.
    - Identifies dependencies among attributes and appropriate identifiers/keys for entity sets and relations (evaluated by a refined ERD diagram and normalized tables, and SQL / process taken to normalize)
    - Justifies the quality of a schema through a theoretical lens (Proposed, not sure how this is to be done, and will consult TA)
* Back-end Engineering
  + Not planned yet