**Introduction to Databases**

1. Define the term “Database” as it relates to this course.
   1. Databases are an organized collection of data. Relational databases are where information is stored in different tables that all relate to each other in some way (i.e. MS Excel).
2. Identify the two central components of a database.
   1. Data
   2. Meta Data
3. Define the acronyms DBMS and RDBMS.
   1. Database Management System
   2. Relational Database Management System
      1. Database Management Software consists of:
         1. Data definition language
         2. Data manipulation language
         3. Data control language
         4. Query Language (i.e. SQL)
         5. Proprietary language tools
         6. Database Engine
4. Describe the purpose of the major components of a DBMS.
   1. DML is all about making changes to the data.
   2. QL is all about pulling out or getting a chance to see the information that’s inside the database.
   3. DCL is about controlling access to the database (i.e. who can modify the data, etc.).
5. List the advantages of using a database technology.
   1. Using database technology is more secure and accurate, allowing you to store data and easily update more data when needed, as well as easily updatable and maintainable among many people.
      1. Summary:
         1. Size
         2. Ease of Updating
         3. Accuracy
         4. Security
         5. Redundancy
         6. Data is Important
6. List the steps of a database design process.
   1. Talk to users (i.e. what data are we storing, what purpose does it serve?)
   2. Review source documents (i.e. how is the user storing data beforehand?)
   3. Identify:
      1. Data items
      2. Business rules around working with all this information
   4. Create a logical database design (logical meaning “big picture”)
7. Define a computer program.
   1. A computer program is a set of instructions for manipulating information.

**Entity Relationship Diagrams**

1. Define the term ERD.
   * Entity Relational Diagram defines data and gives a name to it..
2. Identify the parts of an ERD.
   * Variable is the logical part of the ERD
   * Datatype is the Physical part of the ERD
3. Three classifications of primitive information.
   * Numerical information (represented by numbers)
   * Textual information (represented by words)
   * Conceptual information (represented by either numbers or words)
4. Define the terms Entity, Attribute and Relationships as they relate to databases.
   * **Entity** is the table in the database. **Attributes** are the items which translate into columns in the table (i.e. “Customer” is the table name, “FirstName, LastName, Address”, are attributes that exist within the “Customer” table). **Relationships** are the relationships between different tables (i.e. “Customer, Order, Item, Order Details” – refer to pictures taken in class).
5. Describe the difference between atomic and composite attributes
6. Describe the difference between stored and derived attributes
   * Derived attributes are numbers that can be calculated, stored attributes can’t (i.e. GST and quantity are stored, total amount is calculated).
7. Explain the purpose of a primary key in a database table
   * Unique identifier for each row of data.
8. Define the term technical key
   * Designer “inserts” the technical key to replace a concatenated key.
9. Define the term concatenated key (also known as a composite key)
   * A concatenated key is a primary key with two or more attributes.
10. Define the term Foreign Key
    * A foreign key is an attribute that is also a key attribute in another table (i.e. “CustomerNumber” is a primary key in “Customer”, while also being another attribute in the “Order” table).
11. Define the term cardinality.
    * Cardinality allows us to see what sort of relationship is shared between tables (i.e. the three types of cardinality – quantitative relationship between entities).
12. List the three major types of cardinality.
    * + - One-to-one
        - One-to-many
        - Many-to-many
13. Identify primary keys, foreign keys, and indexes on an ERD.
    * **Primary Key** will act as a key identifier for each row of data (i.e. “Customer Number” is the primary key for the customer). **Foreign Keys** relate an attribute from one table that is also a primary key in another. **Indexes** are attributes that be referenced within a table (i.e. the “Phone” attribute is an index in the “Customer” table. So to find a customer, we could find them by looking up their phone number).
    * Primary Keys act, by definition, as clustered indexes. All other indexes are non-clustered.
14. Identify the cardinality indicated by the relationship lines on an ERD.
15. Describe how to translate an Entity Relationship Diagram into English using a template pattern.
    * Each [**First Entity**] must be (1)/may be (0) [**relationship –** prepositional phrase] one and only one (1)/one or more (many) [**Second Entity**] (i.e. Each customer may be placing one or more orders **or** each order must be placed by one and only one customer – refer to pictures taken in class).
      + - Each order must be made up of one or more order details.
        - Each order detail must be associated with one and only one order.
        - Each order detail must be a sale of one and only one item.
        - Each item may be on one or more order details (refer to pictures taken in class).

**Normalization**

1. List the reasons why we go through the process of "normalizing" metadata
   * Reduce likelihood of errors or anomalies when going through the contents of the database.
     + Create – Inserting
     + Read – Querying
     + Update – Adding
     + Delete – Removing
2. Identify the "normal forms" up to and including 3NF
   * 0NF – List out metadata (in atomic form)
   * 1NF – Separate out repeating groups
   * 2NF – Separate out partial dependencies
   * 3NF – Separate out transitive dependencies
3. Describe the "normal forms" up to and including 3NF
   * Give a name to the complete form (this is our core entity)
   * List all the metadata as atomized attributes
   * Identify the unique primary key attribute
   * Identify which attributes represent repeating data
4. Analyze source documents to distinguish between meta-data and data
5. Analyze a form to identify metadata (0NF) and create a list of the metadata
6. Remove repeating groups, if any, by isolating them into their own distinct entity (1NF) (while maintaining relationships)
7. Check for partial dependencies, if any (2NF)
8. Check for transitive dependencies, if any (3NF)
9. List key questions to ask yourself when checking whether you've correctly processed meta-data through 1NF to 3NF.

**Generate ERDs from 3NF**

1. Draw an ERD diagram to represent your final set of 3NF meta-data
2. Translate your ERD diagram into English so that you can verify your logical analysis of the database with your client
3. Merge ERD diagrams from various views/forms into a single, cohesive logical ERD
4. Identify when to introduce technical keys into your normalized entities
5. Use Visio to draw ERD diagrams
6. Identify good practices for laying out entities in an ERD

**Keep it All Straight**

* When you give an entity a name, *keep that same name* each time you modify the list of attributes in that entity.
* When you name your attributes in **0NF**, *key the same names* for these attributes. If you are placing those attributes in a new entity, *do* ***not*** rename the attributes.
* ***Never*** change the primary key from a previously established entity. (Note: There are some exceptions to this rule, but as a beginner, changing a primary key is a strong indicator that you’re doing it wrong.)
* *Never* change the foreign keys from previously established entities when rewriting them as modified entities.
* Always self-check the relationship between your entities at each step of the normalization process. Make a quick note of the one-to-many relationships that should form as you create new entities eat each step; the cardinal relationship between the entities should make sense.