Assignment #2
Business Database Design

Purpose: To demonstrate the ability to identify functional dependencies and anomalies, and to apply normalization priniciples to nonrelational tables.

Problem: The initial database design was reviewed, which resulted in noting a few issues or concerns. Because normalization addresses redundancy and modification issues, the proposed database design needs to be reevaluated. This process is critical because it will ensure that all functional dependencies and anomalies have been identified and addressed appropriately, which will lead to well formed relations.

Instructions: Using your proposed solution in Assignment #1, follow the normalization process to ensure that the relations are well formed. Implement the normalized database and prepare a report that considers the following:

- ☐ Discuss and explain how the relations were normalized. In other words, what approach was used as part of normalizing the database design and how were the comments/feedback integrated in the process for developing well formed relations.

- ☐ Define each of the relations using the parenthetical method. Each relation in your database design must be a normalized relation (3NF). Please make sure to state your assumptions. An example of the parenthetical method for a "STUDENT" relation is:

  STUDENT (<u>StudentId</u>, StudentName, Email, Age, Major)

  *Note: Attributes which comprise the primary key should be <u>underlined</u> and attributes that are foreign keys should be *italized*.

- ☐ For each relation, create a matrix that provides the fields, data types, field size, and description (if applicable). In addition, any constraints that need to be considered should be noted underneath each matrix.

- ☐ Using your knowledge of SQL DDL, create the script to implement the database tables in SQL Server. Reference the 3NF relations (step 2) and the matrix (step 3) as a guide to define the properties and structure in SQL syntax. Do not forget to enforce referential integrity between the tables by defining PK/FK constraints. (Include the SQL syntax in the report).

  *Note: SQL syntax is written and saved in a text editor application (e.g., Notepad).

- ☐ Execute the SQL script. When you are finished implementing your database tables, create a database diagram in SQL Server Management Studio (SSMS) by following these steps:
  - Right-click on the Database Diagrams item under your database in the Object Explorer
  - Select "New Database Diagram."
  - In the "Add Table" dialog box which appears, select all of the tables in your solution and then click "Add" button.
  - When your diagram is complete, select "Copy Diagram to Clipboard" from the SSMS "Edit" menu. Paste your diagram into the document that will be the final report.

☐ Enter the “dummy data” from Assignment #1, plus an additional 5 records in each table for a total of 10 records. Use SQL DDL to create the syntax in a text editor and try to have multiple records versus one single record where appropriate. (Include the SQL syntax in the report).

☐ Execute SQL statements and provide screenshots that illustrate the “dummy” records of each table.

☐ Create a single document that merges all of the above into one file. **Save and submit as a pdf file.**

*Note: Only one submission per team is needed.