Assignment 5 Part 1

The customers table relates to the orders table through the customer_id key, a one-to-many relationship. The orders table relates to the order_items table through the order_id key, a one-to-many relationship. The products table relates to the order_items table through the product_id key, a one-to-many relationship. The categories table relates to the products table through the category_id key, a one-to-many relationship. The customers table relates to the addresses table through the customer_id key, a one-to-many relationship. For each of these one-to-many relationships, one table can have multiple values from the table that acts as the “many” in the relationship. For example, one customer can have multiple addresses. On the other hand, each part of the “many” in the relationship can only have one value from the “one” in the relationship. In this case, while each customer can have multiple addresses, each address can only have one customer. Customers to addresses has a cardinality of one-to-many.

Assignment 5 Part 2

I achieved first order of normalization by splitting the first and last name of the student into separate columns. This ensures that every data element is stored in its own separate column. I achieved second order of normalization by giving a unique identifier to majors and classes. The student table already had a unique identifier, so I did not need to create one for that table. I achieved third order of normalization by storing the student majors and student classes as relationships to avoid data duplication. The student to major relationship is a many-to-one, while the student to class relationship is a many-to-many. The student to class relationship table has a composite primary key to prevent duplicate student class entries.