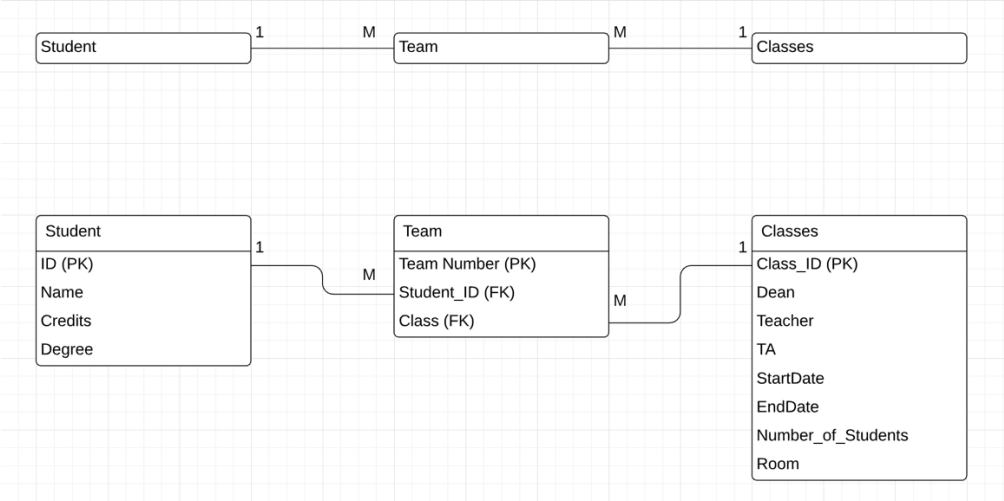
Section 2: Database Design

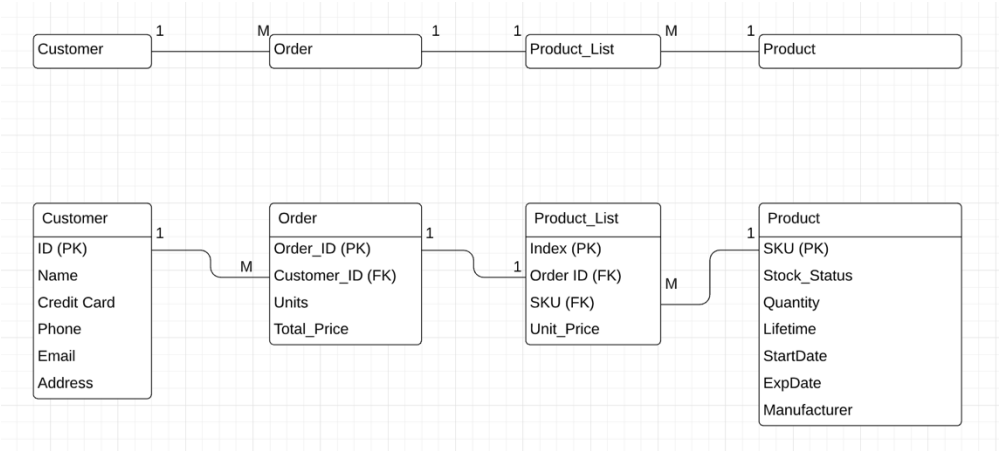
1. **Students M: M Classes**

   As a result of many-to-many relationship, we should transform it into 1-to-1 relationship. Learning from Hult, I added Team entity. Team will have Team Number as the Primary key in team table. Moreover, Student_ID and Class will consider as Foreign Keys to connect the two tables.



2. **Customers M: M Products**

   Similar to the above relationship, we transform it into 1-to-1 relationship. However, I added 2 more tables: Order and Product_List. In Order table, it will show amount of units. When users dig into Order_ID, they could access to each unit of the specific order and the related price. SKU will be mentioned in Product_List as the Foreign Key, additionally; the SKU is then used as Primary Key in Product table.



3. **ER diagram for a library reservation system for a family of libraries based on the given characteristics.**

We designed the database as followed:

Table of Borrowers with Borrower_ID as Primary Key, the table will contain information about the borrowers such as name, address, email and phone number. Similarly, I create tables of library, book and content. As all the relationships at this time are M: N, I create Request table to transform these relationships into 1: M. At this stage, we could achieve the fundamental requirements that multiple types of content that can be borrowed, borrow multiple items at the same time, borrow multiple types of content.

Furthermore, we could dig into local borrowers and book available in the local library. I created two more tables: Local_Borrowers to transform M: N between Borrowers and Library, as a result it is 1-to-1. Additionally, I create the Book_Available table to make Book and Library relationship M: N become 1-to-1.