**Your database assignment includes.**

* Describe the difference between a relational database and a NoSQL database.
  + Relational: Relational databases are essentially gigantic spreadsheets storing all datapoints for particular items uniformly.
  + NoSQL: Whereas NoSQL stores information however it likes under a single header.

For the following questions assume it is a relational database.

* Provide an explanation of the reason for needing a database rather than a spreadsheet (flat file), to keep track of **customers, products, and orders setup.**
  + A database is required here because this is could potentially be an endless list of information. A single spreadsheet could not hold all this information or process it affectively. A database would be able to split the load of information and make it accessible by customer, product, and orders tables; all of which being stored across multiple partitions if capacity warrants.
* Provide details of the data you need to store in each of the tables. Include data types and length.
  + customers
    - Unique identifier/customer # - Integer
    - First name - String
    - Last name - String
    - Address - String
    - Citi - String
    - State - String
    - Zip Code - Integer
    - Birthday(?) – Integers combined to a string
    - Other misc. information relevant to available products
  + products
    - Product identifier
      * By name - String
      * By number - Integer
    - Price - Integer
    - Number of items available - Integer
    - Discount value(?) - Integer
    - Images(?) – IMAGE
  + orders
    - order numbers - Integer
    - ordered date – Integers combined to string
    - order received date - Date
    - Item(s) ordered – Integer to connect to product
    - number of items ordered - Integer
    - cost of item(s) - Integer
    - customer information – Integer to connect to unique customer ID
* Provide details of why and how you split the data into tables.
  + The reason for splitting all this information would be to have unique ways to identify the different information across many different tables. I would this by having a single unique identifier in each table that attaches all the information in each table to 1 piece of information that can be called and pulled to other tables. While any piece of information can be pulled, there can be multiple people with the same name, or multiple similar kinds of products, or multiple orders that are for the same number of products or types of products. For this reason, you keep the single unique characteristic that can be used later to pull up the more specific information.
* What is a relationship between two database tables?
  + A relationship between two database tables is where unique identifiers is used to from one table to pull to another. Meaning, when Customer A places an order for Product C, it would generate a relationship in the orders table showing that Customer A Ordered Product C all in one spot with its own unique identifier that could be used in a different table. This could continue on for whatever information you might want.
* Describe the difference between a one-to-many relationship and a one-to-one relationship. Give examples.
  + A one-to-many relationship is providing a single piece of information from one tablet that can show up multiple times in another table. For example, having a list of customers in a table with all the orders they have placed.
  + A one-to-one relationship is sending one piece of information from one table that will only show up once in the new table. For example, having a tablet that has all of the unique customer ID’s with the total number of orders they have placed.
* Produce a database diagram. ERD - Entity Relational Diagram. Include the tables, all of the fields, the relationships between the tables, primary keys, foreign keys, all of the data fields in each table.

Graphical user interface, application

Description automatically generated

* What is SQL?
  + SQL is a query language used most effectively with relational databases. It allows you to pull any information from any table however you would like to see it. (I used this a lot in my Target days)
* What would the SQL command be if you wished to display the rows of the customer table?
  + SELECT \* FROM CUSTOMER