Consider a database that describes customers, orders and products.  Customers place orders for products with specific delivery dates.  The company then delivers products on dates that may or may not meet the customer's requirement.  Products have prices.  An order can be placed for several products at once.  The order has a total price for all the products ordered, as well as a shipping cost and a total cost.  Products each have a price and a quantity available.  Orders cannot be placed for more units of a product than are available.

1. Construct a data model for this problem and submit a Chen-style ERD.

ERD

A diagram of a customer order

Description automatically generated

Relational Schema

A diagram of a customer id

Description automatically generated

1. Use ERDPlus to generate SQL statements to create the tables and create them in Oracle.  Submit the SQL generated by ERDPlus and evidence that the tables were successfully created by Oracle.

**SQL Scripts**

CREATE TABLE CUSTOMERS

(

Customer\_ID INT NOT NULL,

First\_Name INT NOT NULL,

Last\_Name INT NOT NULL,

Address INT NOT NULL,

Phone\_Number INT NOT NULL,

Email INT NOT NULL,

PRIMARY KEY (Customer\_ID)

);

CREATE TABLE ORDERS

(

Order\_ID INT NOT NULL,

Date\_Requested INT NOT NULL,

Total\_Price INT NOT NULL,

Shipping\_Cost INT NOT NULL,

Quantity\_Requested INT NOT NULL,

Customer\_ID INT,

PRIMARY KEY (Order\_ID),

FOREIGN KEY (Customer\_ID) REFERENCES CUSTOMERS(Customer\_ID)

);

CREATE TABLE PRODUCTS

(

Price INT NOT NULL,

Delivery\_Date INT NOT NULL,

Quantity\_Available INT NOT NULL,

Product\_ID INT NOT NULL,

PRIMARY KEY (Product\_ID)

);

CREATE TABLE Forthe

(

Quantity INT NOT NULL,

Order\_ID INT NOT NULL,

Product\_ID INT NOT NULL,

PRIMARY KEY (Order\_ID, Product\_ID),

FOREIGN KEY (Order\_ID) REFERENCES ORDERS(Order\_ID),

FOREIGN KEY (Product\_ID) REFERENCES PRODUCTS(Product\_ID)

);

**Created Tables**

A screenshot of a computer

Description automatically generated

A screenshot of a black and yellow screen

Description automatically generated

1. Use APEX to construct a simple application that will display each table and allow the user to edit the tables.

**Application**

A screenshot of a computer

Description automatically generated

1. Show two different ways to model the relationship between an employee and the employee's manager for an organization where employees are organized into sections, each section is part of a department, and each department is part of a division.  The divisions make up the company.  What are the relative advantages and disadvantages of the two types of data models?

The two ways to model the relationship:

**Aggregation Model:**

It is a pattern of containment within one another. There are multiple levels to this. According to this model, the corporation will be the primary entity, with numerous divisions, departments, and managers who are also employees inside each division and department.

**Advantages:**

Reduces redundancy.

Reduces data inconsistency

**Disadvantages:**

Retrieving data involves joining multiple tables, making it slower

**⁠Single Entity type Model:**

One table containing the information on each employee's boss, department, and division will be included in this model.

**Advantages:**

Easy to understand.

Only table to store all the information.

**Disadvantages:**

Not scalable.

Creates redundancy.

1. Add type tables to your ERD from problem 1.  Submit the new ERD.

A diagram of a company

Description automatically generated

1. What is the primary key of a type table?  Why?

The primary key of the type table is same as the main table. The primary key of customer type is customer ID, Order type is Order ID and product type is Order ID.

1. Design an ERD for a database of students, where Student is a supertype.  The subtypes are Undergraduate, Graduate and Research Assistant.  A student must be either an undergraduate or a graduate, and a student may or may not be a research assistant.  Submit a Chen-style ERD for this data model.

A diagram of a student

Description automatically generated