**Database Design Project**

**CSCI 210**

**Chris Kiger**

**4/19/21**

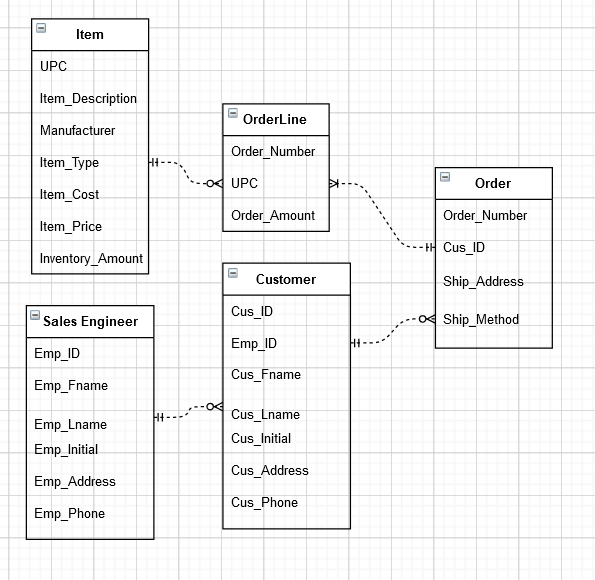
**1. Introduction**

Sweetwater Sound is a distributor of instruments and other sound equipment that stores its inventory in its on-site warehouse. They buy their inventory from hundreds of manufacturers and allow customers to buy everything they need at one location rather than having to contact each of these manufacturers individually. They have a sales force to help customers find the right equipment for their needs. They use a database to keep track of inventory, customers, orders, and employees. They also use the database to keep track on when to order more inventory and what amounts to order.

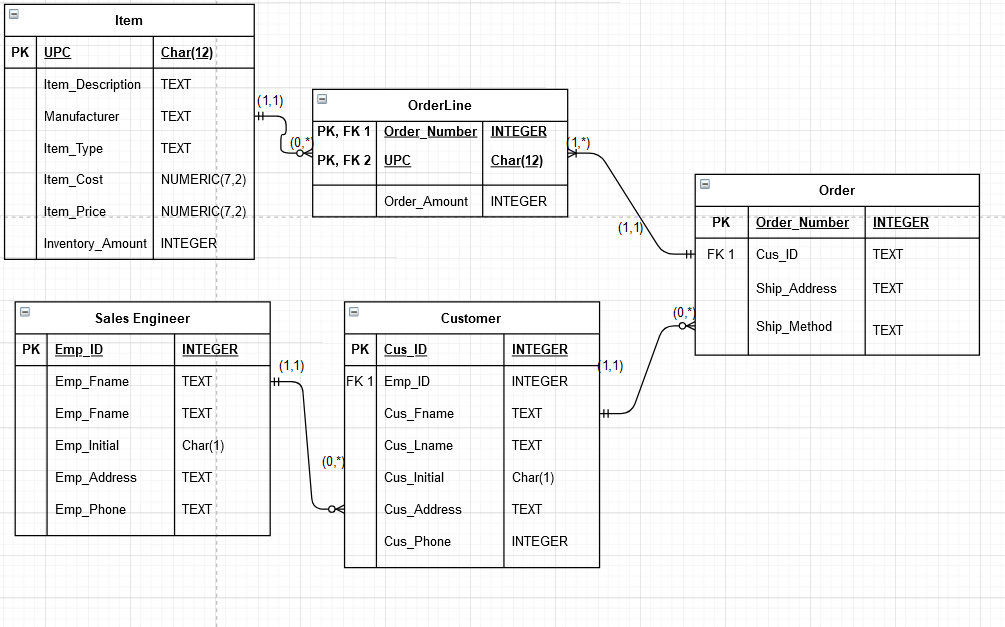
**2. Business Rules**

1. Each customer has a sales rep assigned to them
2. Each inventory item has a sales price and cost
3. Orders are comprised of items
4. The total sales price and cost of the order is equal to the sum of its item’s price/cost
5. Orders are assigned to only one customer
6. Orders may have any number of items
7. Each order has one shipping method
8. Items must exist in inventory to be ordered
9. Sales Engineers are paid based on the total price sold in all orders attached to them

**3. Entity Relationship Diagram**



**4. Entity Relationship Diagram (Expanded)**



**5. Primary Key Table**

|  |  |
| --- | --- |
| **Entity** | **Primary** Key |
| Sales Engineer | Emp\_ID |
| Customer | Cus\_ID |
| Order | Order\_Number |
| Orderline | Order\_Number, UPC |
| Item | UPC |

**6. Integrity Rules**

***Entity Integrity Rule*** – The primary key must be unique and it may not be null.

***Referential Integrity Rule*** – The foreign key may be null (if it isn’t part of the table’s primary key), but it must have a valid entry.

|  |  |  |  |
| --- | --- | --- | --- |
| **Entity** | **Entity Integrity** | **Referential Integrity** | **Comments** |
| Customer | The Cus\_ID is a unique (surrogate) PK and it will not be null. | The customer entity contains one FK, Emp\_ID. It is PK for its table and will be valid. | Request fulfills entity integrity. Request fulfills referential integrity because the FK will be valid. |
| Sales\_Engineer | The Emp\_ID is a unique (surrogate) PK and it will not be null. | The Sales\_Engineer entity does not contain a FK. | Product fulfills entity integrity and does not have a FK for referential integrity. |
| Item | The UPC is a unique PK and it will not be null. | The Item entity does not contain a FK. | Product fulfills entity integrity and does not have a FK for referential integrity. |
| Order | The Order\_Number is a unique (surrogate) PK and it will not be null. | The Order entity contains one FK, Cus\_ID. It is PK for its table and will be valid | Request fulfills entity integrity. Request fulfills referential integrity because the FK will be valid. |
| Orderline | The PK for the Orderline is a composite key composed of Order\_Number and UPC. As PK of other tables, they will not be null | The event entity contains two FK’s –  Order\_Number and UPC. They are both PK’s for their tables and will be valid. | Event fulfills entity integrity. Event fulfills referential integrity because the two referenced FK’s cannot be invalid. |

**7. Definition of Relationships Table**

|  |  |  |  |
| --- | --- | --- | --- |
| Entity 1 | Entity 2 | Relationship | Cardinality |
| Item | Orderline | One Item may exist on zero to many Orderlines. Orderlines will always have 1 Item associated | 1:M |
| Order | Orderline | One Order may be associated with one to many Orderlines. Orderlines will always have 1 Order associated | 1:M |
| Customer | Order | One customer may have zero to many Orders.  Orders are placed by one customer. | 1:M |
| Sales\_Engineer | Customer | One Sales\_Engineer may have zero to many Customers.  Customers will always have 1 Sales Engineer | 1:M |

**8. Database in SQLite** – Attached

**9. SQL Statements used in DB Browser**

BEGIN TRANSACTION;

CREATE TABLE IF NOT EXISTS "Item" (

"UPC" char(12),

"Item\_Description" TEXT,

"Manufacturer" TEXT,

"Item\_Type" TEXT,

"Item\_Cost" NUMERIC(7, 2) NOT NULL,

"Item\_Price" NUMERIC(7, 2) NOT NULL,

"Inventory\_Amount" INTEGER DEFAULT 0,

PRIMARY KEY("UPC")

);

CREATE TABLE IF NOT EXISTS "Sales\_Engineer" (

"Emp\_ID" INTEGER,

"Emp\_Fname" TEXT NOT NULL,

"Emp\_Lname" TEXT NOT NULL,

"Emp\_Initial" char(1),

"Emp\_Address" TEXT NOT NULL,

"Emp\_Phone" INTEGER NOT NULL,

PRIMARY KEY("Emp\_ID" AUTOINCREMENT)

);

CREATE TABLE IF NOT EXISTS "Customer" (

"Cus\_ID" INTEGER,

"Emp\_ID" INTEGER,

"Cus\_Fname" TEXT NOT NULL,

"Cus\_Lname" TEXT NOT NULL,

"Cus\_Initial" char(1),

"Cus\_Address" TEXT NOT NULL,

"Cust\_Phone" INTEGER NOT NULL,

PRIMARY KEY("Cus\_ID" AUTOINCREMENT),

FOREIGN KEY("Emp\_ID") REFERENCES "Sales\_Engineer"("Emp\_ID")

);

CREATE TABLE IF NOT EXISTS "Order" (

"Order\_Number" INTEGER,

"Cus\_ID" INTEGER,

"Ship\_Address" TEXT NOT NULL,

"ship\_Method" TEXT NOT NULL,

PRIMARY KEY("Order\_Number" AUTOINCREMENT),

FOREIGN KEY("Cus\_ID") REFERENCES "Customer"("Cus\_ID")

);

CREATE TABLE IF NOT EXISTS "Orderline" (

"Order\_Number" INTEGER,

"UPC" char(12),

"Order\_Amount" INTEGER NOT NULL,

PRIMARY KEY("Order\_Number","UPC"),

FOREIGN KEY("Order\_Number") REFERENCES "Order"("Order\_Number"),

FOREIGN KEY("UPC") REFERENCES "Item"("UPC")

);

**10. Primary Keys in SQL Statements – highlighted in teal.**

**11. Foreign Keys in SQL Statements – highlighted in pink.**

**12. Database Populated with Five Records per Table**

INSERT INTO Item (UPC, Item\_Description, Manufacturer, Item\_Type, Item\_Cost, Item\_Price, Inventory\_Amount)

VALUES (111222333444, "Bose Stereo", "Bose", "Stereo", 50.00, 124.99, 100);

INSERT INTO Item (UPC, Item\_Description, Manufacturer, Item\_Type, Item\_Cost, Item\_Price, Inventory\_Amount)

VALUES (222333444555, "Fender Guitar A-10", "Fender", "Guitar", 350.00, 799.99, 18);

INSERT INTO Item (UPC, Item\_Description, Manufacturer, Item\_Type, Item\_Cost, Item\_Price, Inventory\_Amount)

VALUES (333444555666, "Fender Guitar 76ST", "Fender", "Guitar", 550.00, 1024.99, 15);

INSERT INTO Item (UPC, Item\_Description, Manufacturer, Item\_Type, Item\_Cost, Item\_Price, Inventory\_Amount)

VALUES (444555666777, "Fender Guitar Y76A", "Fender", "Guitar", 950.00, 2099.99, 18);

INSERT INTO Item (UPC, Item\_Description, Manufacturer, Item\_Type, Item\_Cost, Item\_Price, Inventory\_Amount)

INSERT INTO Sales\_Engineer (Emp\_Fname, Emp\_Lname, Emp\_Initial, Emp\_Address, Emp\_Phone)

VALUES ("Miguel", "Riviera", NULL, "123 Rodeo Drive", 5556473489);

INSERT INTO Sales\_Engineer (Emp\_Fname, Emp\_Lname, Emp\_Initial, Emp\_Address, Emp\_Phone)

VALUES ("Colin", "Peterson", NULL, "456 Brix Road", 5557895145);

INSERT INTO Sales\_Engineer (Emp\_Fname, Emp\_Lname, Emp\_Initial, Emp\_Address, Emp\_Phone)

VALUES ("Tyler", "Davis", "L", "789 Brine Way", 5551679534);

INSERT INTO Sales\_Engineer (Emp\_Fname, Emp\_Lname, Emp\_Initial, Emp\_Address, Emp\_Phone)

VALUES ("Travis", "Sanchez", "E", "234 Torres Pass", 5557824931);

INSERT INTO Sales\_Engineer (Emp\_Fname, Emp\_Lname, Emp\_Initial, Emp\_Address, Emp\_Phone)

VALUES ("Korbin", "Conley", NULL, "567 Heaven Street", 5552468135);

INSERT INTO Customer (Emp\_ID, Cus\_Fname, Cus\_Lname, Cus\_Initial, Cus\_Address, Cust\_Phone)

VALUES (1, "Tommy", "Bugati", "A", "987 Leavenworth Avenue", 5557315984);

INSERT INTO Customer (Emp\_ID, Cus\_Fname, Cus\_Lname, Cus\_Initial, Cus\_Address, Cust\_Phone)

VALUES (1, "Billy", "Idol", NULL, "654 Border Pass", 5557456824);

INSERT INTO Customer (Emp\_ID, Cus\_Fname, Cus\_Lname, Cus\_Initial, Cus\_Address, Cust\_Phone)

VALUES (1, "Sam", "Moore", NULL, "321 Fun Way", 5551264785);

INSERT INTO Customer (Emp\_ID, Cus\_Fname, Cus\_Lname, Cus\_Initial, Cus\_Address, Cust\_Phone)

VALUES (2, "Brandi", "DePrey", NULL, "876 Stix Street", 5553571594);

INSERT INTO Customer (Emp\_ID, Cus\_Fname, Cus\_Lname, Cus\_Initial, Cus\_Address, Cust\_Phone)

INSERT INTO "Order" (Cus\_ID, Ship\_Address, ship\_Method)

VALUES (1, "987 Leavenworth Avenue", "Fedex 2Day");

INSERT INTO "Order" (Cus\_ID, Ship\_Address, ship\_Method)

VALUES (3, "321 Fun Way", "Fedex Priority");

INSERT INTO "Order" (Cus\_ID, Ship\_Address, ship\_Method)

VALUES (4, "876 Stix Street", "UPS Ground");

INSERT INTO "Order" (Cus\_ID, Ship\_Address, ship\_Method)

VALUES (1, "987 Leavenworth Avenue", "USPS Priority");

INSERT INTO "Order" (Cus\_ID, Ship\_Address, ship\_Method)

VALUES (1, "987 Leavenworth Avenue", "Fedex Ground");

INSERT INTO Orderline (Order\_Number, UPC, Order\_Amount)

VALUES (1, 555666777888, 1);

INSERT INTO Orderline (Order\_Number, UPC, Order\_Amount)

VALUES (2, 111222333444, 2);

INSERT INTO Orderline (Order\_Number, UPC, Order\_Amount)

VALUES (3, 555666777888, 5);

INSERT INTO Orderline (Order\_Number, UPC, Order\_Amount)

VALUES (4, 555666777888, 6);

INSERT INTO Orderline (Order\_Number, UPC, Order\_Amount)

VALUES (5, 444555666777, 13);