

Chris Hall & John McCormick

Instructor: Dr. Danielle Safonte, M.I.S. Ed.D., & Dr. Michael Curry

CS340 Introduction to Databases

11 May, 2023

### **Project Step 3 Draft Version: Design HTML Interface + DML SQL**

## **Team Members' Names and Roles**

Chris Hall and John McCormick  
Team 86, Schema-Teama

## **Project Title**

Portland Cycling Specialties

## **Link to HTML**

<http://flip1.engr.oregonstate.edu:7821/>

## **Fixes based on Feedback from Step 2:**

The feedback we received was helpful in thinking about the structure of our database overall. There seems to be confusion by one person about the need for two different prices for a product. Since one price is consumer-facing and the other is the price the bike shop spends to acquire the product functionality of the database around it will not be changed. One other person suggested removing auto\_increment from orderID, which we have done. Additionally, it seems that it's hard for people to follow along with our sample data in this document since the columns were not titled. This has been fixed to make it easier to read.

- Removed AUTO\_INCREMENT from orderID.
- Updated sample data to have column titles.

## Project Outline and Database Outline - Updated Version:

### Overview

Portland Cycling Specialties was a small bicycle specialties shop. They were a small mom and pop business which has grown in the past decade into multiple locations. Sales are up to more than \$1500 in 200 sales on average each day in most of their stores and the old manual system of recordkeeping no longer works well for them. They need a good database system in which to track sales at each location and for orders which come in on their website.

The database will need to track customers and their orders, as well as inventory at store location, and products and suppliers.

### Database Outline in Words

1. **Customers:** Records of each customer who makes a purchase.
  - customerID: int, auto\_increment, unique, not NULL, PK
  - firstName: varchar(45), not NULL
  - lastName: varchar(45), not NULL
  - email: varchar(45), not NULL
  - address: varchar(45), not NULL
  - phoneNum: varchar(45), not NULL
    - i. One-to-many (1:M) relationship with Orders using customerID as a foreign key in Orders.
2. **Products:** Items we sell at Portland Cycling Specialties.
  - productID: int, auto\_increment, unique, not NULL, PK
  - name: varchar(45), not NULL
  - description: varchar(45), not NULL
  - price: decimal(10,2), not NULL
  - supplierID: int, not NULL, FK (ref Suppliers)
    - i. One-to-many (1:M) relationship with Inventory using productID as a foreign key in Inventory.
    - ii. Many-to-many (M:N) relationship with Orders implemented in the OrderProducts table.
3. **Suppliers:** Companies which supply products to us.
  - supplierID: int, auto\_increment, unique, not NULL, PK
  - name: varchar(45), not NULL
  - email: varchar(45), not NULL
  - phone: varchar(45), not NULL
  - representativeName: varchar(45), not NULL

- i. One-to-many (1:M) relationship with Products using supplierID as a foreign key in Products.

**4. Inventory:** Quantity of each product at each of our locations.

- inventoryID: int, auto\_increment, unique, not NULL, PK
- productID: int, not NULL, FK (ref Products)
- locationID: int, not NULL, FK (ref Locations)
- inStockQuantity: int, not NULL
  - i. One-to-many (1:M) relationship with Products using productID as a foreign key in Inventory.
  - ii. One-to-many (1:M) relationship with Locations using locationID as a foreign key in Inventory.
  - iii. foreign key in Inventory.

**5. Orders:** Records of sales to each customer.

- orderID: int, unique, not NULL, PK
- customerID: int, not NULL, FK (ref Customers)
- orderDate: date, not NULL
- total: decimal(10,2), not NULL
  - i. One-to-many (1:M) relationship with OrderProducts using orderID as a foreign key in OrderProducts.
  - ii. One-to-many (1:M) relationship with Customers using customerID as a foreign key in Orders.

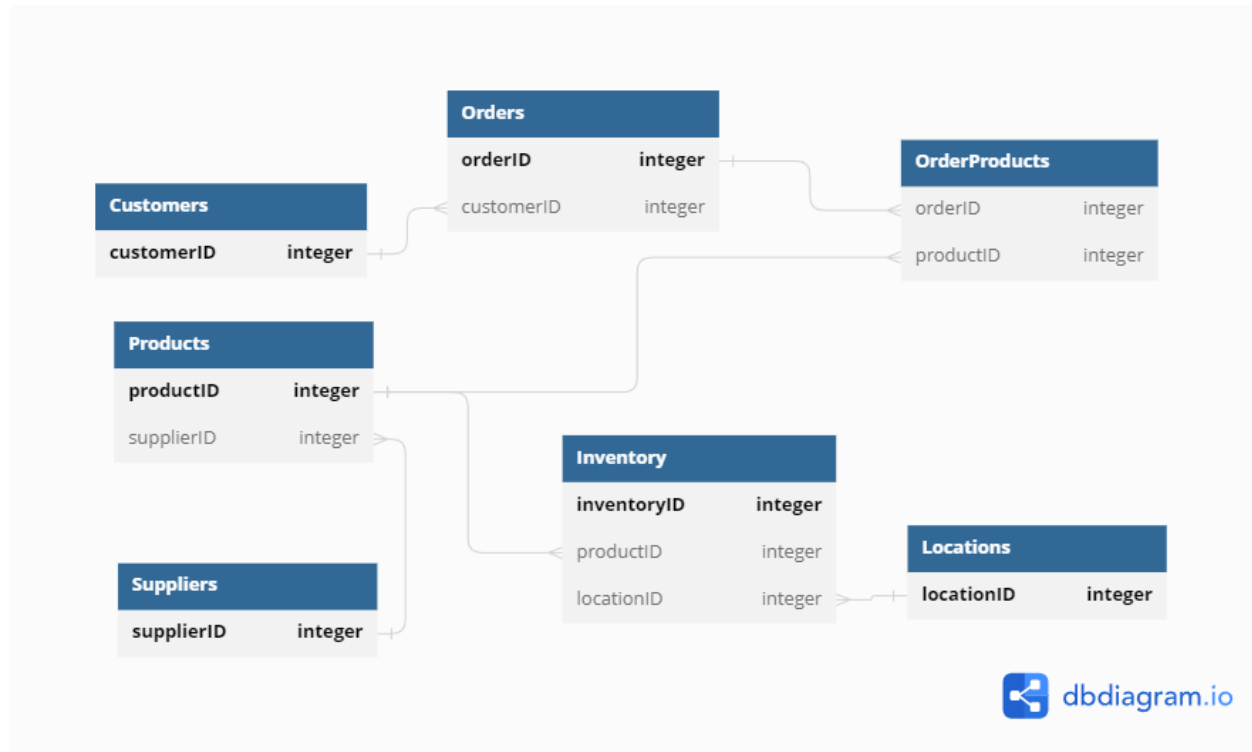
**6. OrderProducts:** Links Orders and Products in a M:N relationship.

- orderID: int, not NULL, FK (ref Orders)
- productID: int, not NULL, FK (ref Products)
- quantity: int, not NULL
- price: decimal(10,2), not NULL
  - i. Many-to-many (M:N) relationship with Products implemented using productID and orderID as foreign keys.

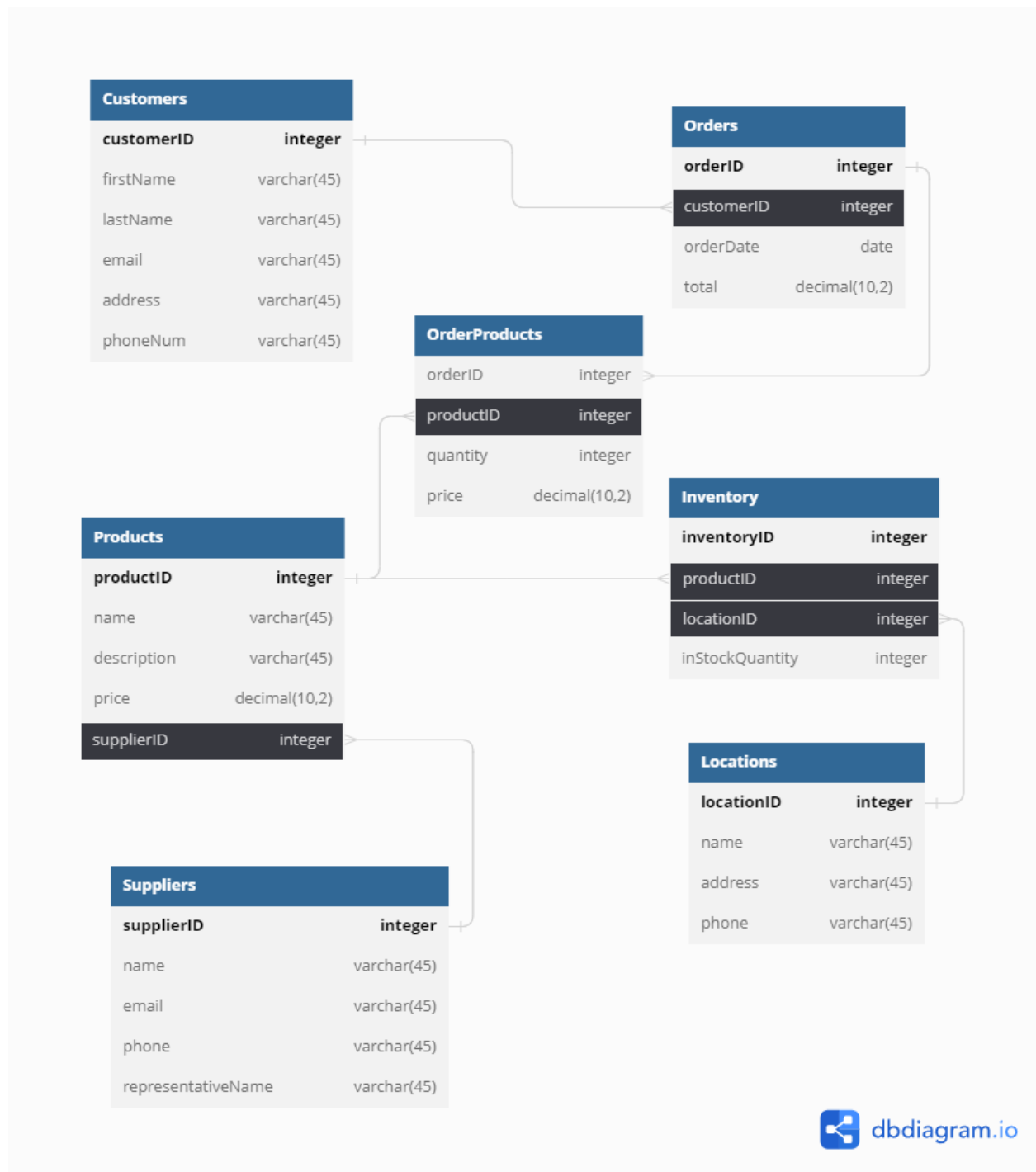
**7. Locations:** Our company stores locations.

- locationID: int, auto\_increment, unique, not NULL, PK
- name: varchar(45), not NULL
- address: varchar(45), not NULL
- phone: varchar(45), not NULL
  - i. One-to-many (1:M) relationship with Inventory using locationID as a foreign key in Inventory.

## Database Diagram



## Schema



## Example Data

### Customers:

customerID	firstName	lastName	email	address	phoneNum
1	John	Doe	j.doe@email.com	123 Main St, Portland	5035555555
2	Jane	Doe	jane.d@email.com	456 Elm St, Portland	5035555556
3	Bob	Dole	b.dole@email.com	789 Oak St, Portland	5035555557

### Products:

productID	name	description	price	supplierID
1	Mountain Bike	Sturdy all-terrain bike	350.00	1
2	Road Bike	Lightweight and fast road bike	550.00	2
3	Helmet	Protective headgear	50.00	3

### Suppliers:

supplierID	name	email	phone	representativeName
1	AllTerrain Co.	allterrain@email.com	5035550001	John
2	SpeedBikes Inc.	speedbikes@email.com	5035550002	Sarah
3	SafeCyclist Ltd.	safecyclist@email.com	5035550003	Robert

### Inventory:

inventoryID	productID	locationID	inStockQuantity
1	1	1	10
2	2	1	8
3	3	1	20

**Locations:**

locationID	name	address	phone
1	Portland Downtown	124 Main St, Portland	5035551122
2	Portland East	456 82nd St, Portland	5035552233
3	Gresham	789 Main St, Gresham	5035553344

**Orders:**

orderID	customerID	orderDate	total
1	1	2023-05-01	700.00
2	2	2023-05-02	600.00
3	3	2023-05-03	400.00

**OrderProducts:**

orderID	productID	quantity	price
1	1	1	350.00
1	3	1	50.00
2	2	1	550.00
3	3	2	100.00