

Comp 430/533

Assignment 4

A4: Building & Populating our Database

The goal of this assignment is to actually construct on possible version of the database we designed in Assignment 3. As a result, you will become an expert on our application and our business.

What to turn in

You must turn in a .sql file on Canvas. Table definitions should be specified in the order in which they can be successfully defined (for example, if the saleDetail table references the sale table, be sure to define the sale table first). Basically, we want to be able to hit execute and run your sql code to create your tables. The order in which the tables are described here may not be a “run-able” ordering. This also means that any comments or text answers in your file should be in SQL comments.

Grading

Building and loading the tables is worth 40 points, total. Yes, it takes a while, but it’s not particularly difficult to do. Creating the INSERT statements for the specified data is worth 30 points. Short answer questions are worth a total of 10 points. Queries are worth a total of 20 points.

What’s In and Out of Scope

This is intended to be a declarative SQL assignment. Therefore, you must write queries in SQL (not functions). You may use VIEWS as needed and you may use standard built-in PostgreSQL functions (e.g. ROUND, IF or CASE statements). If you’re not sure if something is allowed, ask!

Academic Honesty

The following level of collaboration is allowed on this assignment:

You may discuss the assignment with your classmates at a high level. Any issues getting PostgreSQL running is totally fine. What is not allowed is direct examination of anyone else's SQL code (on a computer, email, whiteboard, etc.) or allowing anyone else to see your SQL code.

You may use the search engine of your choice to look up the syntax for SQL commands, but may not use it to find answers.

You MAY post and discuss query results with your classmates.

1 Create Tables

Create the following tables for our ice cream food truck. Provide your SQL code. Be sure to specify NULL or NOT NULL, any appropriate default values, primary and foreign keys, and appropriate constraints.

Each table MUST have a PRIMARY KEY defined. Sometimes, you will need to decide the appropriate attribute(s) that form the PRIMARY KEY.

Note: “number” in these tables refers to a to-be-specified numeric type field. You need to determine which exact data type should be used.

1.1 Product

Attribute Name	Attribute Type
productCode	CHAR(3)
productName	VARCHAR(50)
isAvailable	INTEGER

1.1.1 Constraints




1. Product codes must be unique

2. Product names must be unique
3. productCode is the primary key
4. productName must be specified
5. by default, isAvailable is set to 0


1.2 ProductPrice

Attribute Name	Attribute Type
productCode	CHAR(3)
startDate	DATE
cost	number
price	number


1.2.1 Constraints


1. the combination of productCode and startDate form the primary key
2. productCode must be a value in the product table 
3. cost and price are not required when the record is created 
4. cost and price are currency values 

1.3 Ingredient


Attribute Name	Attribute Type
ingId	SERIAL INTEGER 
ingName	VARCHAR(50)
category	VARCHAR(50)

1.3.1 Constraints


1. ingId is the primary key
2. ingName may be repeated 

3. the combination of ingName and category must be unique 
4. All of the attributes must be specified


1.4 Recipe

Attribute Name	Attribute Type
productCode	CHAR(3)
ingId	INTEGER
qty	number 
unit	VARCHAR(20)

1.4.1 Constraints

1. Each ingredient may be listed in a recipe at most one time 
2. productCode must be a value in the product table
3. qty must be specified
4. unit must be specified
5. ingId must be a value in the ingredient table

1.5 truckEvent

Attribute Name	Attribute Type
eventId	SERIAL
eventName	VARCHAR(200)
eventStart	date and time
plannedEnd	date and time 
actualEnd	date and time

1.5.1 Constraints

1. eventId is the primary key.
2. eventName must be specified and does NOT need to be unique

3. eventStart is required
4. plannedEnd is not required
5. actualEnd is not required
6. the combination of eventName and eventStart must be unique

1.6 Sale

Attribute Name	Attribute Type
saleId	SERIAL
eventId	INTEGER
productCode	CHAR(3)

1.6.1 Constraints

1. saleId is the primary key
2. eventId must be a value in the truckEvent table
3. productCode must be a value in the product table

1.7 SaleDetail

Attribute Name	Attribute Type
saleId	INTEGER
ingId	INTEGER
qty	number
unit	VARCHAR(20)

1.7.1 Constraints

1. the combination of saleId and ingId form the primary key
2. saleId must be a value in the sale table
3. ingId must be a value in the ingredient table

4. qty and unit must be specified

1.8 Supplier

Attribute Name	Attribute Type
supplierId	SERIAL
supplierName	VARCHAR(150)
street	VARCHAR(150)
city	VARCHAR(150)
state	VARCHAR(50)
postalCode	VARCHAR(15)
country	VARCHAR(150)

1.8.1 Constraints

1. supplierId is the primary key
2. the combination of supplierName, street, and city must be unique

1.9 SupplierPhone

Attribute Name	Attribute Type
supplierId	INTEGER
phoneType	VARCHAR(50)
phoneNumber	VARCHAR(20)

1.9.1 Constraints


1. the combination of supplierId and phone type form the primary key

1.10 LocalSupplier

Attribute Name	Attribute Type
supplierId	INTEGER

Attribute Name	Attribute Type
mileageCost	number
distance	number

1.10.1 Constraints

1. supplierId is the primary key
2. the supplierId value must exist in the supplier table
3. both mileageCost and distance must be able to accommodate non-integer values and must be specified
4. mileageCost may contain fractions of cents (e.g. 54.5 cents) 

1.11 NationalSupplier

Attribute Name	Attribute Type
supplierId	INTEGER
transportFee	number

1.11.1 Constraints

1. supplierId is the primary key
2. the supplierId value must exist in the supplier table

1.12 ingQuote

Attribute Name	Attribute Type
quoteId	SERIAL
supplierId	INTEGER
issueDate	DATE
expirationDate	DATE
tax	number
fees	number

Attribute Name	Attribute Type
total	number

1.12.1 Constraints

1. quoteId is the primary key
2. the supplierId value must exist in the supplier table
3. issueDate must be specified
4. expirationDate must be specified
5. the combination of supplierId and issueDate must be unique
6. tax, fees, & total must handle currency values without losing precision
7. tax and fees must be specified
8. total will be calculated later 📄

1.13 quoteItem

Attribute Name	Attribute Type
quoteId	INTEGER
ingId	INTEGER
qty	number
unitCost	number
unit	VARCHAR(20)



1.13.1 Constraints

1. records are uniquely identified by quoteId and ingId
2. the quoteId value must exist in the quote table
3. the ingId value must exist in the ingredient table
4. qty, unitCost, and unit must all be specified
5. unitCost must handle currency values

1.14 Delivery

Attribute Name	Attribute Type
deliveryId	SERIAL
quoteId	INTEGER
orderDate	DATE
deliveryDate	DATE


1.14.1 Constraints

1. deliveryId is the primary key
2. the quoteId value must exist in the quote table
3. orderDate is the date our ice cream truck order decided to purchase the items on the quote. To be considered a “delivery” this field must be populated 
4. deliveryDate will start off empty and be populated later, when the order actually arrives 

1.15 inventoryItem

Attribute Name	Attribute Type
ingId	INTEGER
quoteId	INTEGER
expirationDate	DATE
stockQty	number
unit	VARCHAR(20)
qtyRemaining	number

1.15.1 Constraints

1. The primary key is determined by ingId and quoteId
2. expirationDate is only populated for ingredients with a short shelf life (e.g. fruit) 

3. stockQty and unit must be populated
4. the quoteId value must exist in the quote table
5. the ingId value must exist in the ingredient table
6. qtyRemaining will start as stockQty and be decremented as we consume inventory. For now, it can be NULL.

1.16 Equipment

Attribute Name	Attribute Type
equipmentName	VARCHAR(50)
installDate	DATE

z

1.16.1 Constraints

1. each piece of equipment is assigned a unique name that also serves as the primary key
2. the install date must be specified

1.17 Maintenance

Attribute Name	Attribute Type
maintId	SERIAL
equipmentName	VARCHAR(50)
description	VARCHAR(150)
beforeFlag	INTEGER
triggerQty	INTEGER
triggerUnit	VARCHAR(20)
minutes	INTEGER

1.17.1 Constraints

1. maintId is the primary key
2. equipmentName value must exist in the equipment table
3. beforeFlag, triggerQty, triggerUnit, and duration must be specified

1.18 MaintenanceLog

Attribute Name	Attribute Type
maintId	INTEGER
datePerformed	DATE
minutes	INTEGER
notes	TEXT

1.18.1 Constraints


1. records are uniquely identified by the combination of maintId and maintDate

2 Create Data




Write and submit INSERT statements for the scenarios described below. Again, the order in which you list your INSERT statements matters, and we must be able to run your INSERT commands in the order you specify.

2.1 db sundae

Provide INSERT statements for a product called “db sundae”, its ingredients, price, and recipe. We will sell a “db sundae” for \$4.

This product has productCode “db” and is composed of the following ingredients 

- 1 “10 oz dish” (category “cup”)

- 1 “tall napkin” (category “paper goods”)
- 6 ounces of ice cream base   
- 1.5 ounces of a topping
- 1 “short spoon” (category “spoon”)
- 1.5 ounces of hot fudge topping

Possible toppings are: “sprinkles”, “oreo”, and “peanuts”.

Possible ice cream flavors are: “chocolate” and “vanilla”.

Also create products for

- ProductCode “sx” - which is an “extra sundae topping”. This product’s cost is 5 cents, the price is 25 cents, and it consists of 1.5 ounces of a topping.
- ProductCode “mt” - which is a “monkey tail”. (it’s a chocolate covered frozen banana). This product’s price is \$5 and it costs \$2.50.
- ProductCode “dk” - which is a “drink”. This product’s price is \$0.75 and it costs \$0.44. Customers can choose from “coke”, “sprite”, and “water”.

2.2 Equipment

Our ice cream truck has two ice cream machines, named “Rice” and “Owl”. Each machine must be cleaned once per week. It takes 120 minutes to clean each machine. We also have a generator that must be refueled after 40 hours of operation. Refueling takes 10 minutes. The generator also needs its oil & filter changed after 200 hours of operation.

Write SQL statements that define this equipment and maintenance.

Write additional statements indicating

- The generator was refueled on March 1, 2018 and the following note was made: “Purchase more diesel!”.

- “Rice” was cleaned on March 2, 2018, but the cleaning only took 100 minutes.
- “Owl” was also cleaned on March 2, 2018 and the cleaning took 110 minutes.

2.3 TruckEvent

Our ice cream truck was at an event named “RMC study break” on March 1, 2018. The event was scheduled from 8-11 PM but actually ran until 11:10 PM.

At the event we sold 53 “db sundae”s. Of these, 25 were vanilla and the remainder were chocolate. 10 sundaes had no toppings at all, 20 had oreos, 3 had peanuts and the remainder had sprinkles.

We also sold 5 monkey tails, 10 cokes, 22 sprites and 37 bottles of water.

Create INSERT statements for these sales. Arbitrarily assign flavors & toppings based on the above specifications. You may generate this data any way you like (python code, rolling a die, etc.).

2.4 Quotes

Our ice cream truck purchases ingredients from 4 different suppliers. (Note all suppliers and locations are fictitious).

- Local supplier: “Houston’s Best Food” located at: 934 University Blvd, Houston, TX 77005, USA; Transportation fee \$7.
- Local supplier: “Local Premium Food” located at: 101 Main St, Houston, TX 77004, USA Transportation fee \$5.
- National supplier: “Best Food in Canada” located at: 735 First Ave., Toronto, Ontario, M4B 1B5, Canada; Distance 1,530 miles. Mileage cost \$0.10.
- National supplier: “LA Ice Cream Supply” located at: 535 King St., Lake Charles, LA, 70601; Distance 148 miles. Mileage cost \$0.11.

We receive a quote for the following items from Houston's Best Food:

- Strawberries 20 pounds \$2 / pound
- Vanilla ice cream 5 gallons \$3.00 / gallon
- Chocolate ice cream 5 gallons \$3.50 / gallon
- The quote is issued on February 2, 2018 and expires on February 10, 2018.
- tax = \$5.98
- fees = \$7
- total = sum of all item costs + tax + fees

and a competing quote from Local Premium Food:

- Strawberries 25 pounds \$1.75 / pound
- Vanilla ice cream 5 gallons \$3.20 / gallon
- Chocolate ice cream 5 gallons \$3.45 / gallon
- The quote is issued on February 3, 2018 and expires on February 9, 2018.
- tax = \$6.35
- fees = \$5
- total = sum of all item costs + tax + fees

Create SQL INSERT statements to populate the ingQuote and quoteItem tables for these quotes.

3 Short answer questions

3.0.1 Short answer 1 (3 points)

What different number types did you use? Why did you choose those?

3.0.2 Short answer 2 (2 points)

Category is a string in the ingredient table. Provide 1 advantage and 1 disadvantage of storing the category name in the ingredient table.

3.0.3 Short answer 3 (1 points)

If we didn't have maintId in the maintenance table, what could we use instead?

3.0.4 Short answer 4 (3 points)

What changes would need to be made to our database if we purchased a second truck?

3.0.5 Short answer 5 (1 point)

My favorite flavor of ice cream is:

4 Queries

Answer all of the questions below by writing and executing SQL queries. The queries must contain ONLY the answer to the question (no extra rows or columns). You may only use SQL to answer the questions.

4.0.1 Query 1(5 points)

What is the recipe for the “db sundae”? Include the productCode, productName, each ingredient id, ingredient name, category, quantity and unit. Sort by ingredient name.

4.0.2 Query 2 (10 points)

Write a SQL statement to determine the quote with the lowest cost.

4.0.3 Query 3 (5 points)

Write a query that returns the total income (sum of the prices of all the products sold) we collected during the GSA study break event. While we currently only have one event and sales data in our database, your solution should work when there are multiple events and sales data from those events.