Data Warehousing

Homework 2
Grocery Chain Data

Breitzman 1/29/2025

The Idea

- The best way to learn all aspects of Data Warehousing is to actually build a Data Warehouse
- I usually assign people to teams of 2 or 3 but since we have a small class we can have teams or 1, 2, or 3 and still have several grocery stores that we will later roll up into a chain of grocery stores.
- You will modify a program that simulates scanner output for each customer for your store for each day of the year
- Since we want a little variation between the stores, certain parameters will be changed depending on your team

The Idea (II)

- The building of the data warehouse will be based on several of your HW assignments. It sounds like it could be a disaster, but I've done this several times and it usually works well.
- The data warehouse will contain data for a year's worth of sales and inventory data that you will generate.
- This first homework will involve you writing a program to simulate scanner outputs for each customer for your store. (Actually, I have some code from last year that will work, if you don't want to write your own.)
- Later we will use the data to do data cleansing and transformation, as well as generate aggregates, data cubes, business reports, etc.
- For now, we are just generating a year's worth of sales transactions for each store

Basic Assumptions

- We assume our store is open every day of the year (no holidays)
- We run our simulation from Jan 1, 2024 to Dec. 31, 2024
- To keep things simple
  - we will not charge tax
  - Our prices won't change
  - we won't time stamp each customer. We'll just date stamp it and also track a customer number
- This will not be a toy database. Depending on your team you will have 1000 +/- customers per day, and each customer will buy an average of 50 items multiplied by 365 days. So your output might have 10 to 20 million rows.
- Our data warehouse is only modeling 9 stores; data warehouses for Wal-Mart, Home Depot, Target, etc. model thousands of stores and each has more than the thousand customers we have and more than the 2,000 products we have.

Our Stores will Carry About 2,075 Items

- A real grocery store carries about 20,000 items
- Below is a snippet of the pipe delimited text file I will give you

Manufacturer	Product Name	Size	itemType	SKU	BasePrice
American	American Cole Slaw	17.8	Deli Salads	44135001	0.89
Atomic	Atomic Mint Chocolate Bar	9.89	Chocolate Candy	44113001	0.90
BBB Best	BBB Best Creamy Peanut Butter	11.8	Peanut Butter	44051001	1.10
Beech-Nut	Stage 2 Just Apple & Kiwi	4.25 oz	Baby Food	44033001	1.09
Beech-Nut	Stage 1 Beef & Broth	2.5 oz	Baby Food	44018001	1.09
Beech-Nut	Stage 1 Butternut Squash	4.25 oz	Baby Food	44019001	1.09
Beech-Nut	Stage 1 Chicken & Broth	2.5 oz	Baby Food	44020001	1.09
Beech-Nut	Stage 1 Honey Crisp Apples	4.25 oz	Baby Food	44021001	1.09
Beech-Nut	Stage 1 Sweet Potatoes	4.25 oz	Baby Food	44022001	1.09
Beech-Nut	Stage 1 Turkey And Broth	2.5 oz	Baby Food	44023001	1.09
Beech-Nut	Stage 2 Just Apple & Blackberry	4.25 oz	Baby Food	44032001	1.09
Beech-Nut	Stage 1 Bananas	4.25 oz	Baby Food	44017001	1.09
Beech-Nut	Stage 2 Just Apple & Strawberry	4.25 oz	Baby Food	44034001	1.09
Beech-Nut	Stage 2 Just Pear & Blueberry	4.25 oz	Baby Food	44035001	1.09
Beech-Nut	Stage 3 Just Apple & Aronia Berry	4.2 oz	Baby Food	44042001	1.09
Beech-Nut	Stage 3 Mango, Carrot,Strawberry & Chia	4.2 oz	Baby Food	44043001	1.09
Beech-Nut	Stage 2 Apple & Pumpkin	4.25 oz	Baby Food	44024001	1.09
Best Choice	Best Choice No Salt Popcorn	12.6	Popcorn	44009001	1.19
Better	Better Chicken Noodle Soup	11.1	Soup	43988001	1.15
Big City	Big City Canned Mixed Fruit	16.7	Canned Fruit	43959001	1.17
Big K	Citrus Drop Soda	67.6 oz	Soda/Juice/Drinks	43957001	0.89
Big Time	Big Time Frozen Pepperoni Pizza	7.15	Pizza	43946001	1.13
Blue Label	Blue Label Turkey Noodle Soup	13.5	Soup	43927001	0.88
Booker	Booker Cheese Spread	5.42	Cheese	43896001	0.94
Bravo	Bravo Canned Tomatos	11.3	Canned Vegetables	43881001	0.87
Bright And Early	Juice Grape	59 oz		43876001	1.00
Campbells	Spaghetti O's Plus Calcium	14.2 oz	Canned Goods	43856001	1.15
Carlson	Carlson Low Fat Sour Cream	11.1	Sour Cream	43836001	0.81
Carlson	Carlson Havarti Cheese	12.1	Cheese	43831001	1.13
Carrington	Carrington Home Style French Fries	5.86	French Fries	43820001	0.92
Carrington	Carrington Frozen Chicken Breast	8.92	Frozen Chicken	43811001	0.93
CDR	CDR Oregano	13.3	Spices	43798001	1.16
Chef Boyardee	Beefaroni Whole Grain	15 oz		43789001	1.00
Chef Boyardee	Mini Bites Dinos & Meatballs	15 oz		43790001	1.00
Chef Boyardee	Ravioli Beef	15 oz		43791001	1.00
Choice	Choice Tasty Candy Bar	15.2	Chocolate Candy	43783001	0.89
Consolidated	Consolidated Buffered Aspirin	6.56	Aspirin	43746001	0.85
Cutting Edge	Cutting Edge Beef Bologna	4.67	Bologna	43721001	0.97

Your Output will Look Something Like…

Date	Customer #	SKU	Sale Price
20170101	1	44135001	0.98
20170101	1	43610001	1.20
.	.	.	.
.	.	.	.
.	.	.	.
20170101	1111	…	…
.	.	.	.
.	.	.	.
.	.	.	.
20171231	1009	…	…

- Except it will have 10 to 20 million rows (and it will be for the year 2023)

Parameters Based on Your Team

Teams	Daily Customer Count	Weekend Customer Count	Price Multiplier	# items per customer
1,6,11	980-1020	+75	1.1	1 to 60
2,7,12	1000-1040	+75	1.05	1 to 80
3,8,13	1020-1060	+75	1.2	1 to 90
4,9,14	1050-1080	+50	1.07	1 to 70
5,10,15	1070-1100	+50	1.15	1 to 65

We'll figure out who is on what team in a few minutes.

Additional Assumptions

- 70% of your customers will purchase milk
  - Of those that buy milk, 50% will also buy cereal
  - Only 5% of those that don't buy milk, will buy cereal
- 20% of your customers will buy baby food
  - Of those that buy baby food 80% will also buy diapers
  - 1% of customers that don't buy baby food will buy diapers
- 50% of your customers will buy bread
- 10% of your customers will buy peanut butter
  - Of those that buy peanut butter 90% will buy a jam or jelly
  - Of those that don't buy peanut butter, 5% will buy a jam or jelly
- All other products are equally likely (put the SKU's in an array numbered from 0 to N and choose a random number in that range)

Pseudocode

```
Const cCustomersLo = 980
Const cCustomersHi = 1020
Const cPriceMultiplier = 1.1
Const cDate1 = 20170101
Const cMaxItems = 70
Const cWeekendIncrease = 50

For i = 0 to 364 do
        int custCount = randRange(cCustomersLo, cCustomersHi, cWeekendIncrease)
        For j = 1 to custCount do
                int myItems = randRange(1,cMaxItems)
                int k = 0
                If randRange(1,100) <= 70 then /* 70% probability of buying milk */
                        SKU = getMilkSKU()
                        WriteRecord(myDate(i), j, SKU etc.)
                        k++
                        If randRange(1,100) <= 50 then  /* 50% prob of buying cereal */
                                SKU = getCereal()
                                WriteRecord(myDate(i), j, SKU etc.)
                                k++
                Else  /* didn't buy milk. Only 5% prob of buying cereal */
                        If randRange(1,100) <= 5 then
                                SKU = getCereal()
                                WriteRecord(myDate(i), j, SKU etc.)
                                k++
                        End If
                        .
                        . /* Do same for bread, Peanut Butter, Diapers, Other constraints */
                /* now buy (myItems – k) random products */          .
                For m = k to myItems do
                        SKU = getRandomItem()
                        WriteRecord(myDate(i), j, SKU etc.)
                End for
        End For
End For
```

Deliverables

Two Items:

1. Source Code (in any language)
2. Do Not turn in your table with millions of rows. (Save it for the next step) Instead turn in a summary with the following totals (for the 365 day period)
   - Number of customers
   - Total sales
   - Total items bought
   - Top 10 selling items (with product name not just an SKU) with counts

   - You may compute the summaries in a database system if you don't want to complicate your code

Rules of the game

- You may use any language you want: Python, C, C++, C#, Scheme?, Visual Basic, (even Java!)

- Again, you will work in teams (but for this assignment one can take the lead).
- Even if you are on a team with others, everyone should have a submission. (list your teammates if any on the submission.)

- It's not a difficult project (I even gave you some pseudo code and will give you previous classes source code) but it will run for a while, so don't put it off until the last minute.

A means to an end…

- This assignment is a means to an end. I need a bunch of different data-marts to do the rest of the course. So, I don't really want this to be an epic coding exercise

- To speed things along, I am making available code from a previous class. I make no warranties about suitability, so make sure if you lift someone else's code that you test it to make sure it's working properly. Also cite it properly (e.g. I used Mike Leonchuck's code and modified section x and y)

Teams

- Who wants to work alone?
- Who wants to work with others?
- Who wants me to assign them to a team?

2 More Things…

- I've done this a couple of times now. First time it was a bit dodgy and I had to help a lot of people get their code working. At this point, the code works pretty well now, so you really don't have to do much other than get an environment set up. So if you're ready to panic over this, don't.

- If you are someone who always turns in HW late (and you know who you are). Don't do it. We are eventually going to merge all of these disparate databases together into a single warehouse, and if one or more teams are late, it will mess things up