SUPERMARKET DATABASE PROJECT

ANDREW PATILIS, DANNY CHENG,

JAMES GENGERKE, RYAN KINARD

DATABASE DESIGN

- A supermarkets Inventory with
- Departments with Product Details for Items
  - Product Name
  - Product UPC (Universal Product Code)
  - Product Brand
  - Total in Stock per Product
  - Retail Price in USD
  - Wholesale Price in USD
  - Product Origin via the Vendor
- 3rd Normal Form – No Transitive Dependencies

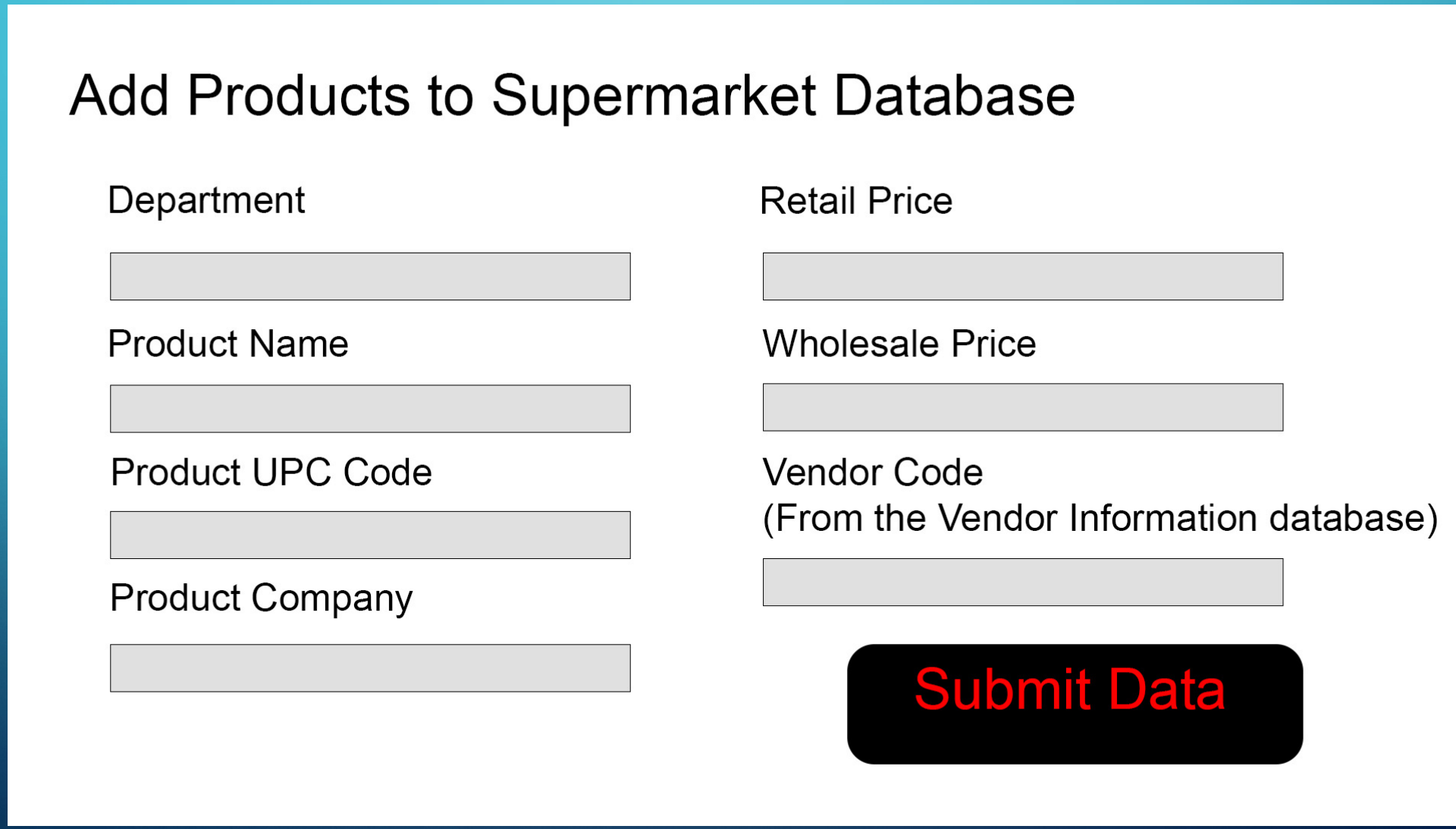
STORY

- Used by a local supermarket or chain of supermarkets
  - Keep track of product and stock
  - Meant for the backend
- Use for checking item price via the UPC code or product name
- Use for checking stock of items via the UPC code or product name

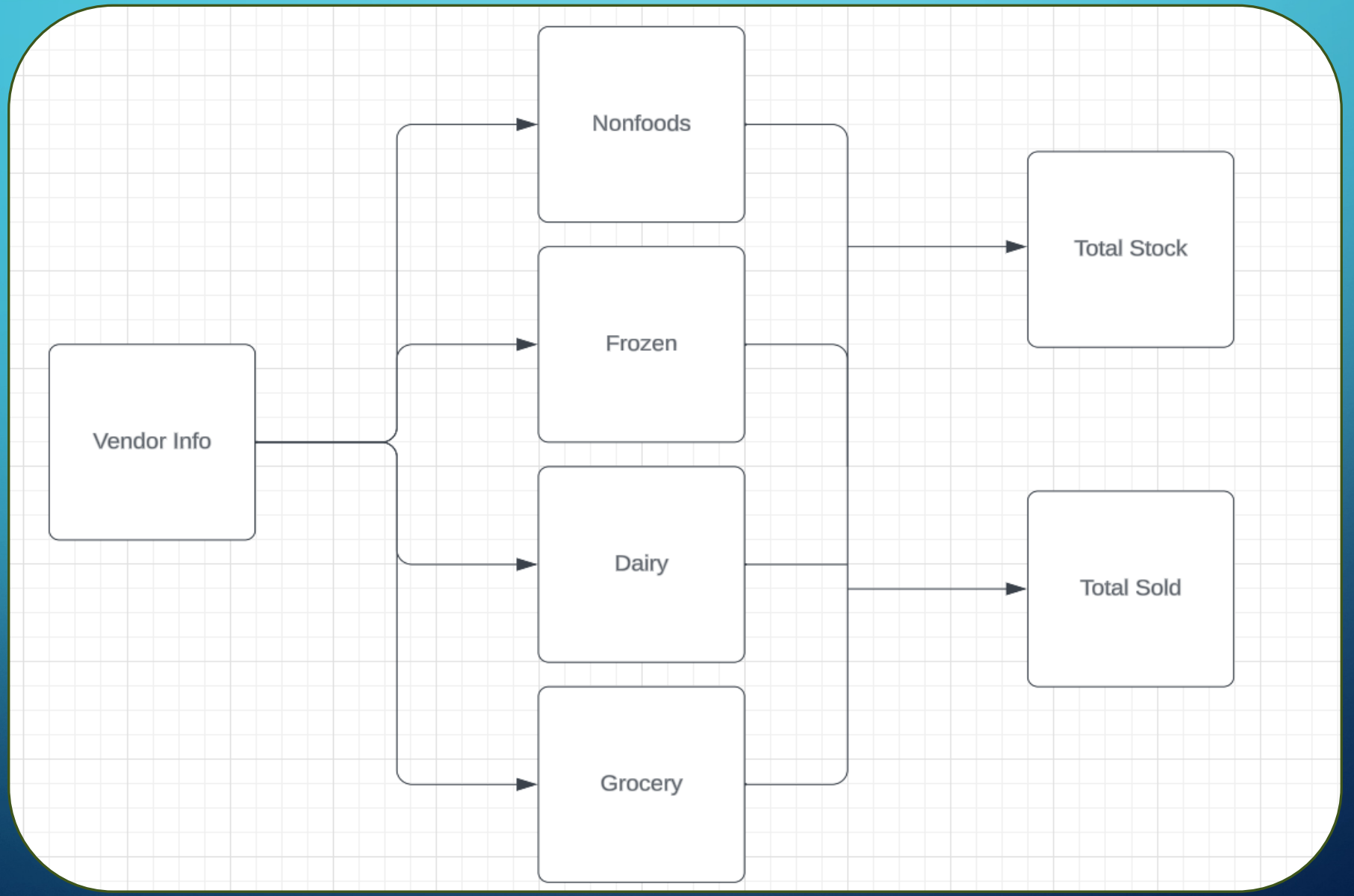
DATA DICTIONARY

- Company: the company who makes the product; in vendorInfo, the companies method of shipping to the retail location in complete pallets.
- Commissary: a warehouse/storage facility where bought product is stored for distribution; often shipped out as mixed pallets.
- productUPC: the UPC (Universal Product Code) of a given product; this is different for every product.
- vendorCode: indicator of a products origin; a Commissary or a Company

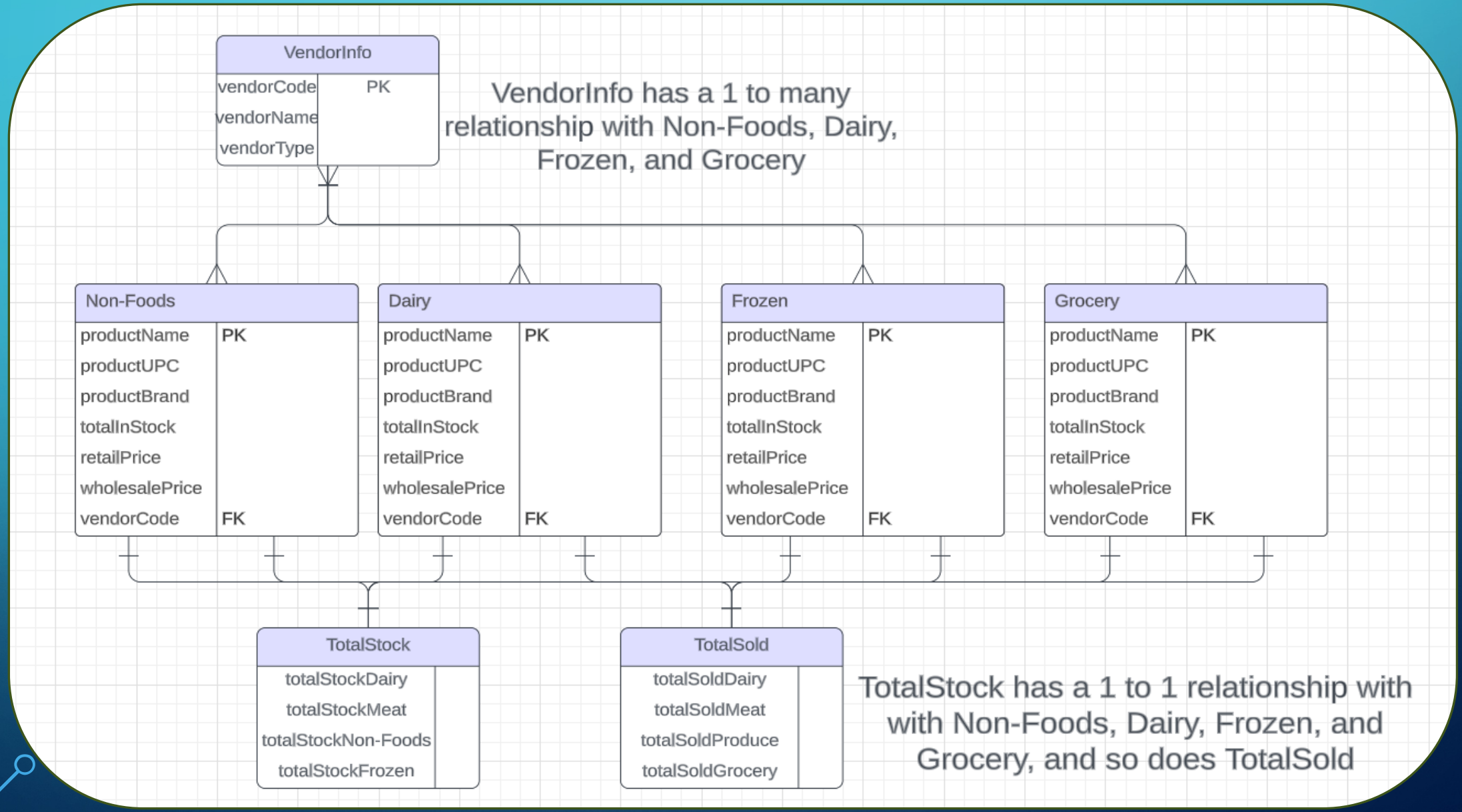
DATA INPUT SCREEN MOCKUP
VALUES –> SQL UPDATE QUERY



ER DIAGRAM


Nonfoods
Frozen
Dairy
Grocery
Vendor Info
Total Stock
Total Sold

MAPPED ER DIAGRAM



SPECIAL TABLES

- Total Sold
  - How many products have been sold over the course of the day in each department.
- Total in Stock
  - How many total products are available for sale in each department.
  - A sum of all in stock items for each department.
- Vendor Code
- Origin of Products
  - Shipped from an organizations Commissary
  - Shipped from an outside company

TOTAL SOLD TABLE

- Shows the amount of product sold in each department for the current day.

department	totalSold
Dairy	498
Frozen	418
Grocery	539
Non-Foods	302

TOTAL IN STOCK TABLE

- Shows the amount of product currently in stock in each department.
- Sum of each products in stock value.

Items	Departments
7108	grocery
340	nonFoods
3045	frozen
747	dairy

VENDOR INFO (VENDOR CODE PRIMARY KEY)

- Origin of Products
- Where the facility is or what company owns it
  - If a company and not a commissary
- Type of ownership
  - company
  - commissary

vendorCode	vendorName	vendorType
1	Keasby	commissary
2	Bayonne	commissary
3	AriZona	company
4	Coca-Cola	company
26	Chattem Inc.	company
91	Advil	company
256	Pictsweet Farms	company
300	Conagra Brands	company

MAIN TABLE EXAMPLE
GROCERY TABLE

productName	productUPC	productCompany	totalInStock	retailPrice	wholesalePrice	vendorCode
Pure Leaf Lower Sugar Subtly Sweet Peach Real Brewed Tea, 16.9 oz, 6 count	00012000213571	Lipton	100	8.29	3.99	1
Pringles Original Potato Crisps Chips, 5.2 oz	00038000138416	Kellogg Company US	328	2.39	0.99	1
Goldfish Mega Bites Sharp Cheddar Baked Snack Crackers, 26.7 oz	00014100053897	Pepperidge Farm	481	11.29	4.99	1
AriZona Green Tea with Ginseng and Honey, 128 fl oz	00613008715120	AriZona	256	3.29	0.99	3
Coca-Cola Cans, 7.5 fl oz, 6 Pack	00049000061017	Coca-Cola	589	5.29	2.99	4
Oreo Cakesters Soft Snack Cakes, 2.02 oz, 5 count	00044000069940	Mondelēz International Foodservice	390	4.19	1.99	1
OREO Chocolate Sandwich Cookies, 13.29 oz	00044000060114	Mondelēz International Foodservice	527	4.19	1.99	1
Polar Toasted Coconut Premium Seltzer, 12 fl oz, 12 count	00071537202885	Polar	74	5.99	2.99	1
Nestlé Toll House Dark Chocolate Morsels, 20 oz	00028000394257	Nestlé	184	6.49	2.99	1
Nestlé Toll House Semi-Sweet Chocolate Morsels, 24 oz	00028000215606	Nestlé	156	6.49	2.99	1

- Stores All of the main data
- Data retrieved from these tables for review

MAIN TABLE SQL

- Create the table
- Insert data using queries similar to the one below
  - Includes many rows for each item
- The insert screen would create a insert query and run that upon submission of the data

```
create table grocery (
    productName varchar(512),
    productUPC varchar(15),
    productCompany varchar(512),
    totalInStock int,
    retailPrice decimal(8,2),
    wholesalePrice decimal(8,2),
    vendorCode int
);
```

```
insert into grocery (productName, productUPC, productCompany, totalInStock, retailPrice, wholesalePrice, vendorCode)
values
('Pure Leaf Lower Sugar Subtly Sweet Peach Real Brewed Tea, 16.9 oz, 6 count', '00012000213571', 'Lipton', 100, 8.29, 3.99, 1),
```

TOTAL STOCK SUMMARY TABLE SQL

```sql
create table totalStock (
Items int,
Departments varchar(255)
);

insert into totalStock (items)
select sum(totalInStock) as groceryTotal from grocery;
update totalStock set Departments=('grocery')
where Items=(select sum(totalInStock) as groceryTotal from grocery);
```

- For each Department, sums the total in stock and displays it with an identifier for the department it retrieved the data from.
- Multiple versions of the second query here for each department.

VENDOR INFO SQL

```sql
create table vendorInfo (
    vendorCode int,
    vendorName varchar(255),
    vendorType varchar(255)
);
insert into vendorInfo (vendorCode, vendorName, vendorType)
values
(1, 'Keasby', 'commissary'),
(2, 'Bayonne', 'commissary'),
(3, 'AriZona', 'company'),
(4, 'Coca-Cola', 'company'),
(26, 'Chattem Inc.', 'company'),
(91, 'Advil', 'company'),
(256, 'Pictsweet Farms', 'company'),
(300, 'Conagra Brands', 'company');
```

- Simple table with the values needed to connect vendorCode to the main tables
- vendorCode in practice would be a foreign key
- This data would need to be input using a separate data entry screen in practice

TOTAL SOLD SQL

```sql
create table totalSold (
department varchar(255),
totalSold int
);

insert into totalSold (department, totalSold)
values
('Grocery', 539),
('Non-Foods', 302),
('Frozen', 418),
('Dairy', 498);
```

- Currently displays set values
- In practice, this would collect sales from Point of Sale systems and display it here as the sales occur during the day; it would be reset daily

CONCLUSION

- Used on the backend of a supermarket
  - Maintain ease of operation
  - Keep track of product
- Does this in 3rd Normal Form
- Stores all necessary data to keep operating effectively