To begin, I created a database in sql named QuantigrationUpdates. This database has three separate tables. The first table has customer information it is named customers and has eight columns that describe customers. The second table stores order information it is named orders and has four columns that describe orders. The last table is named RMA and has five columns that describe rma’s. The next thing that I did was load the data from separate files into the database so I could perform database searches.

I created a search that returned how many orders are from customers in the city of Framingham, Massachusetts. I found that there are 585 orders from customers that live in Framingham, Massachusetts. A search like this can help the company make decisions about staffing at local stores and where future stores can be located. This search can be enhanced by adding information about the orders that were made. That would give the company the insight for which items are selling at this location.

The next search that I performed showed how many customers lived in Massachusetts. From this search I found that 982 customers live in Massachusetts. When combining the information from the previous search of orders in Framingham and the current search of customers that live in Massachusetts, it can be concluded that out of the 982 customers that live in Massachusetts 585 of them live in Framingham. The company can use this information to determine if a physical location would be beneficial in Framingham because a lot of the revenue from Massachusetts comes from Framingham.

Next, I searched for the customers that live in Woonsocket, Rhode Island. I found that only seven customers live in Woonsocket, Rhode Island. This information is useful because the company can make decisions to find out if they should have a store in Woonsocket. This search can be enhanced by adding the additional search to find out how many customers live in Rhode Island.

In addition to searching, I also updated the tables by adding new customers, new orders, and changing the name of the customers table to collaborators. It is also possible to update current records. As well as delete records. My final step was to create an output file that houses all the search results that I performed.

Some limitations that I faced are gaps in the data that I was given. The data would benefit from having a column that included prices in the RMA table. There could also be a column in the customers table that has the amount they spent most recently and the total amount they have spent with the company. In addition, the orders table would benefit from a column that had the amount that was spent for the total order. With the additional columns described more detailed searches can be performed.

Using sql is a great tool for creating databases. But additional searches can be preformed by using a data mining tool. Data mining gives a company more insight into there company and helps to create relationships in the data so the company can analyze and make more informed decisions. The tool that I am going to use today is JMP. I will create several charts that I can analyze to give examples of how this tool can be useful.

Graphical user interface

Description automatically generated

This is a graph that displays the status of the returns in a bar graph. Most of the return statuses are complete. This graph can be enhanced to see the reason for the returns that pending. As well as the reasons for returns that are complete.

Chart

Description automatically generated

Above is a graph of return reasons by the sku of the product. This allows comparisons to be made about the relative frequency of returns.

Chart, histogram

Description automatically generated

This graph shows a representation of the sku numbers that have been ordered. Based on this graph it can be concluded that BAS-48-1C is the sku that is ordered the most.

Let’s pretend the company was beginning to lose revenue. The company wanted to understand why they are losing revenue because sales are continuing to grow. Using the graphs above, an analysis can be made that although sales are rising returns are high. The product that has the highest return frequency is BAS-48-1C. I would suggest the company produce a customer survey; in that survey they should include customers that have purchased the item BAS-48-1C as well as other items. They can then make an informed decision on whether they should keep the item in stock or replace that item with a better item.

The following is my sql searches that I performed to get the information listed above.

CREATE TABLE Customers(CustomerID INT PRIMARY KEY, FirstName VARCHAR(25), LastName VARCHAR(25), Street VARCHAR(50), City VARCHAR(50), State VARCHAR(25), ZipCode INT, Telephone VARCHAR(15));

A picture containing text, scoreboard, plaque

Description automatically generated

CREATE TABLE Orders(OrderId INT PRIMARY KEY, CustomerID INT, SKU VARCHAR(20), DESCRIPTION VARCHAR(50));

ALTER TABLE Orders ADD FOREIGN KEY (CustomerID) REFERENCES Customers(CustomerID);

A picture containing text, scoreboard, plaque

Description automatically generated

CREATE TABLE RMA(RMAID INT PRIMARY KEY, OrderID INT, Step VARCHAR(50), Status VARCHAR(15), Reason VARCHAR(15));

ALTER TABLE RMA ADD FOREIGN KEY (OrderID) REFERENCES Orders (OrderId);

A picture containing text, scoreboard, plaque

Description automatically generated

create schema QuantigrationUpdates;

Text

Description automatically generated

CREATE TABLE Orders(OrderId INT PRIMARY KEY, CustomerID INT, SKU VARCHAR(20), DESCRIPTION VARCHAR(50));

ALTER TABLE Orders ADD FOREIGN KEY (CustomerID) REFERENCES Customers(CustomerID);

A picture containing text, scoreboard, screenshot

Description automatically generated

CREATE TABLE RMA(RMAID INT PRIMARY KEY, OrderID INT, Step VARCHAR(50), Status VARCHAR(15), Reason VARCHAR(15));

ALTER TABLE RMA ADD FOREIGN KEY (OrderID) REFERENCES Orders (OrderId);

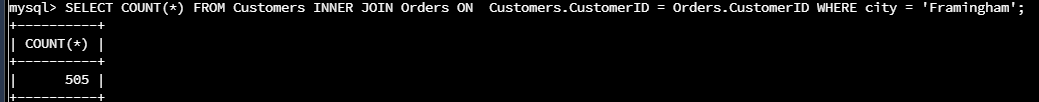
Text

Description automatically generated

CREATE VIEW Collaborator AS SELECT CustomerID AS CollaboratorID FROM Customers;

A picture containing text, scoreboard

Description automatically generated



Text

Description automatically generated

Text

Description automatically generated

Graphical user interface, text

Description automatically generated

Text

Description automatically generated

Graphical user interface, text

Description automatically generated

