HW 3

Title: DB Assignment 3

Name: Brennen Cramp

Date: 10/11/2024

Problem 1:

First, I wanted to combine the 3 tables (merchants, sell, products) with inner joins to be able to get which products are sold by which merchants. A projection is then used to grab the name of the merchant and the name of the product they sell where the product is no longer available (quantity is equal to 0).

A screenshot of a computer

Description automatically generated

Problem 2:

Starting off, I wanted to combine the 2 tables (products and sell) with a left join to be able to get which products are sold via a corresponding merchant ID. A projection is then used to retrieve the name of the product and its description where merchant is not matched to a product thus the merchant ID is null (i.e. meaning that corresponding product is not sold).

A screenshot of a computer

Description automatically generated

Problem 3:

I combined the 5 tables (customers, place, orders, contain, products) with inner joins to be able to get what orders the customers placed and the products that were bought. A projection is then used to grab the count of the customers where they have bought SATA drives and not products with the name “Router.”

A screenshot of a computer

Description automatically generated

Problem 4:

First, I wanted to combine 3 tables (merchants, sell, products) with inner joins to be able to get which products are sold by which merchants. Next, a projection was used to grab the name of the product, quantity available, description of the product, the original price of the product, and I calculated the 20% sale price of the product. Finally, I filter on the networking category of products sold by the HP company.

A screenshot of a computer

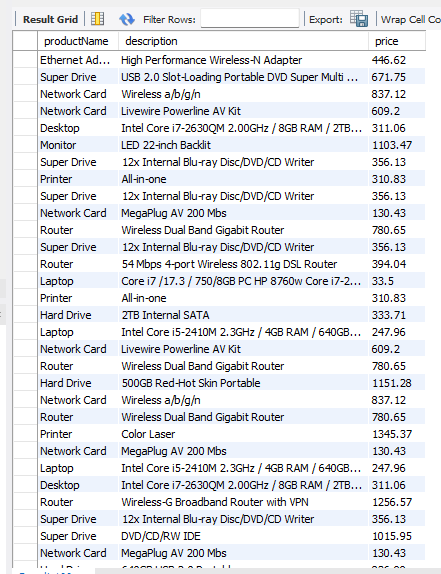
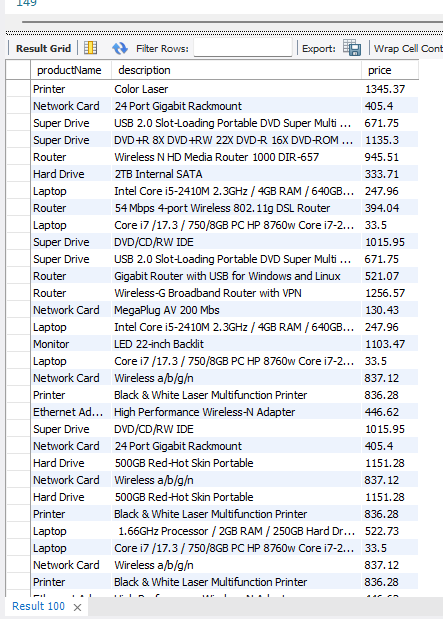
Description automatically generated

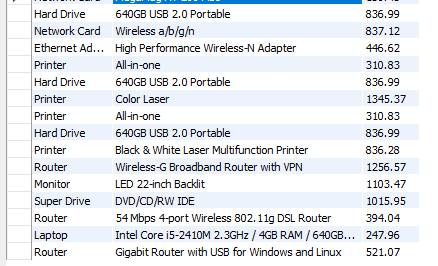
Problem 5:

I first combined 7 tables (customers, place, orders, contain, products, sell, merchants) with inner joins to be able to get all the companies and the customers who have bought products, along with the price of the products bought. Next, I used a projection to grab the name of the product, the product’s description, and the price of the product where I then sort based on the customer’s full name (Uriel Whitney) and the merchants name (Acer) to show me all the products that Uriel Whitney has bought from Acer.

A screenshot of a computer

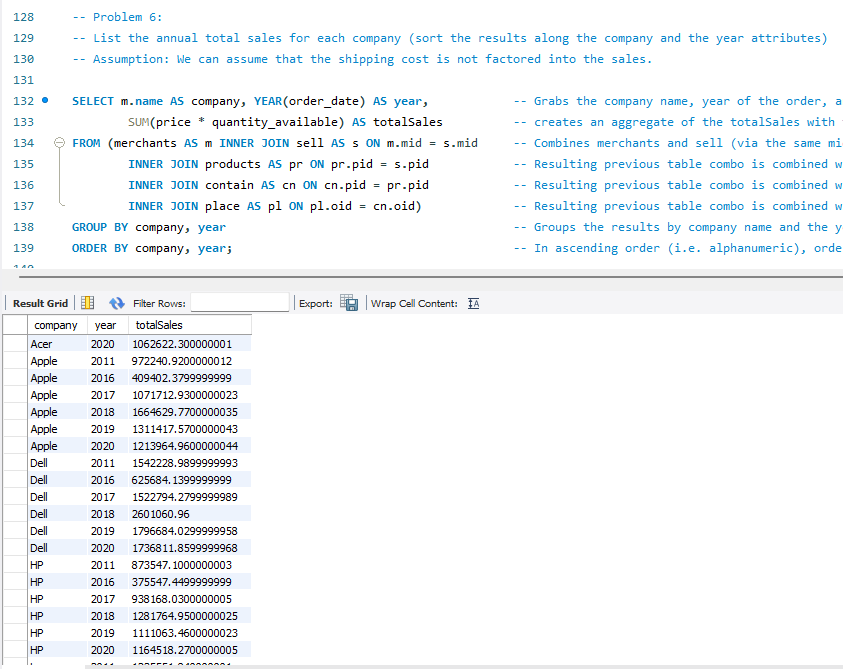
Description automatically generated

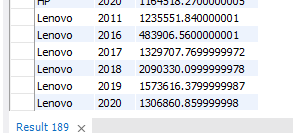




Problem 6:

I combined 5 tables (merchants, sell, products, contain, place) with inner joins to be able to get all the companies and the products that have been sold with their order dates. Next, I used a projection to retrieve the company name, year (using the year() function), and the sum of the price multiplied by the quantity to get the total sales. I also used group by so the sum aggregate can count the total sales by a given company and its corresponding year. Finally, I ordered the results alphanumerically by the company and year.





Problem 7:

I combined 5 tables (merchants, sell, products, contain, place) with inner joins to be able to get all the companies and the products that have been sold with their order dates. Next, I used a projection to retrieve the company name, year (using the year() function), and the sum of the price multiplied by the quantity to get the total sales. I also used group by so the sum aggregate can count the total sales by a given company and its corresponding year. Finally, I ordered the total sales from highest to lowest and used limit 1 to grab the highest annual revenue.

A screenshot of a computer

Description automatically generated

Problem 8:

For this query, I can simply use the orders table where a projection is used to grab the method of shipping and an aggregation to calculate the average shipping cost. I also grouped by the shipping method for the aggregation to find the average shipping cost per the method of shipping and, in ascending order, grabbed the top result off the query to grab the cheapest shipping method (on average) used.

A screenshot of a computer

Description automatically generated

Problem 9:

Using a CTE for the problem, I first combined 5 tables (orders, contain, products, sell, merchants) with inner joins to be able to get all the companies and the best category per company based on its total sold. Next, I used a projection to retrieve the company, category, and sum aggregate of the price multiplied by the quantity available to get the total sold, grouping by the company and category. With that query forming my CTE, my next query is retrieving the company and the highest total from the CTE table where I group the aggregate by company. However, I was stuck on how to grab the category with this query, meaning I am only displaying the company and the total sold.

A screenshot of a computer

Description automatically generated

Problem 10:

Using a CTE for the problem, I first combined 7 tables (customers, place, orders, contain, products, sell, merchants) with inner joins to be able to get all the companies and the customers who have bought products, along with the price of the products bought. Next, I used a projection to retrieve the company, customer name, and sum aggregate of the price to get the total the customer spent, grouping by the company and customer name. With that query forming my CTE, my next queries are retrieving the company and the highest total from the CTE table where I group the aggregate by company; this is then combined using “Union” with the lowest total found in the CTE table when grouping by the company (all done in company alphabetic order). However, like the previous problem, I was stuck on how to grab the customer’s name with this query, meaning I am only displaying the company and the highest and lowest totals spent by the customer at that company. This query and results are shown on the next page:

A screenshot of a computer

Description automatically generated

ERD Diagram:

A screenshot of a computer

Description automatically generated