



SQL Case Study

Worldwide Importers Ltd – Sales Analysis

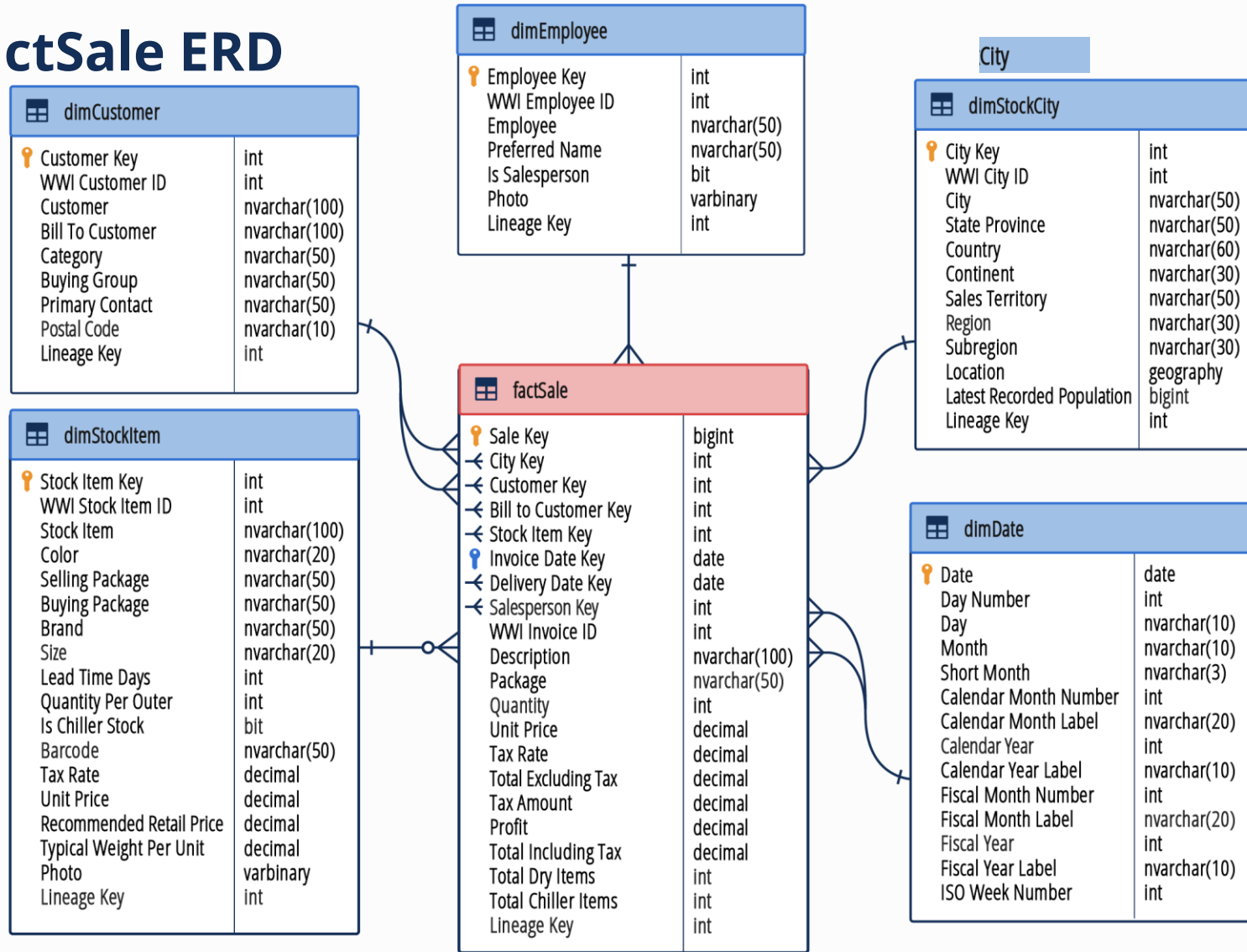
World Wide Importers Scenario

World Wide Importers Inc (WOWI) are a novelty goods importer and wholesale distributor, operating from the San Francisco bay area.

You have joined World Wide Importers as a Business Intelligence Analyst. During your first week, you'll receive questions from several teams across the business. Do what you can to help them.

- Explore the sales database, ERD and other documentation to understand what you're working with.
- Help the marketing team make decisions for a new Halloween social media campaign.
- Help the delivery team make decisions for a delivery efficiency trial.
- Help onboard a new member of the finance team by analyzing data across time.
- Help the product team by evaluating the performance of specific products, including a new chilled products line.

WIWO_DW factSale ERD



Note: There are other tables that form part of this data warehouse, but this case study will focus on these tables.



Basic Queries

Basic Queries - Concepts to Recap

The questions in this section require you to pull data from a single table. You may wish to review the following material from SQL Fundamentals.

The 6 Fundamental SQL Statements – [Link to Chapter From SQL Fundamentals](#)

Aggregate Functions – [Link to lesson from SQL Fundamentals](#)

Count Aggregations – [Link to lesson from SQL Fundamentals](#)

Server Properties – [Link to lesson from SQL Fundamentals](#)

Logical Operators – [Link to lesson from SQL Fundamentals](#)

LIKE Operator – [Link to lesson from SQL Fundamentals](#)

Using CAST to change data type – [Link to lesson from SQL Fundamentals](#)

Using Rollup for Subtotals – [Link to lesson from SQL Fundamentals](#)

Using subqueries – [Link to lesson from SQL Fundamentals](#)

Explore the Database

Connect to the data warehouse using the provided credentials. You have a little time to explore before your first meeting.

How many rows are in the FactSale table?	Easy
What data type is the Quantity column in FactSale?	Easy
How many columns in the FactOrder table can contain null values?	Easy
Does the Staging_factSale table contain any data?	Easy
Do the collation properties of the server suggest the database is case sensitive?	Hard
What does the lineage column mean in the fact and dimension tables?	Medium
What key is enforced between FactSale and dimCustomer dimension?	Easy

Explore the Database

SQL Query	
<pre>SELECT * FROM factSale</pre>	
<pre>SELECT * FROM factOrder</pre>	

Collation

Collation - tells us how our database deals with **case sensitivity** and what **languages** we can use.

SQL Query

```
SELECT CONVERT (varchar(256), SERVERPROPERTY('collation')) AS Collation
```

Timezone

Time zone - important to know for date functions when you use today, now etc.. You can produce a wrong result if you assume the time zone is the same as your computer.

SQL Query

```
SELECT SYSDATETIME() AS SystemTime, GETDATE() AS GetDate
```


Collation

Collation - tells us how our database deals with **case sensitivity** and what **languages** we can use.

- **Latin1_General_CP1** - supports Western European languages. CP1 means code page 1252 and it's an encoding created by Microsoft. It's a very popular one
- **CI** - case insensitive. Size='400L' = Size='400I'
- **AS** - accent sensitive. Once we get to our dim tables, check if some products have accents, if so, we need to be careful when we filter in a where clause for a product that contains an accent.

Basic Customer Data

As with any business, understanding our customers is essential. Answer these questions about our customers.

How would you best describe the type of customers that we have?	Easy
How many rows does the dimCustomer table have?	Easy
Does the customer dimension include an entry for Unknown?	Easy
How many known customers do we have?	Medium

Explore Customer Dimension

SQL Query	
<pre>SELECT * FROM dimCustomer</pre>	Returns full list of customers
<pre>SELECT COUNT([Customer ID]) FROM dimCustomer WHERE [Stock Item] != 'unknown'</pre>	

Stock Item (Product) Questions

You'll soon be working with the marketing team to identify products for a promotion.

In the meantime, get up to speed with the products we sell by exploring the dimStockItem table.

How many foreign keys does the DimStockItem table have?	Easy
How many rows are in the dimStockItem table?	Easy
How many known products are we dealing with?	Hard
What does the "Is Chiller Stock" column mean?	Easy

Exploring Number of Products

SQL Query	
<pre>SELECT * FROM dimStockItem</pre>	Returns all data from the stock item table.
<pre>SELECT COUNT([Stock Item Key]) AS ProductCount FROM dimStockItem WHERE [Stock Item] != 'unknown'</pre>	Counts product numbers excluding those marked as unknown.
<pre>SELECT COUNT(DISTINCT Color) As colorcount FROM dimStockItem WHERE Color <> 'N/A'</pre>	Counts the number of unique colors of products

Marketing Questions

The marketing team are running a new Halloween campaign to reward people who share content on social media. We plan to ship a low cost item to them for free.

What is the unit price of the cheapest product that we currently sell (excluding Unknown Items)? Easy

What is the name of the cheapest product that we currently sell? Medium

Notice that many of the cheapest items relate to packaging materials. Exclude all items that contain the word box, bag, or carton in their names.

What is the cheapest non-packaging related product? Hard

Exploring Cheapest Product

SQL Query`

<code>SELECT MIN([Unit Price]) FROM dimStockItem WHERE [Stock Item Key] !=0</code>	Find the lowest price of any product.
<code>SELECT [Stock Item] AS ProductName, [Unit Price] FROM dimStockItem</code>	Return all products and their prices.
<code>SELECT [Stock Item] AS ProductName, [Unit Price] FROM dimStockItem WHERE [Unit Price] = (SELECT MIN([Unit Price]) FROM dimStockItem WHERE [Stock Item Key] !=0)</code>	Return the name and price of the lowest priced product.
<code>SELECT [Stock Item] AS ProductName, [Unit Price] FROM dimStockItem WHERE [Stock Item Key] !=0 AND [Stock Item] NOT LIKE '%bag%' AND [Stock Item] NOT LIKE '%box%' AND [Stock Item] NOT LIKE '%carton%' ORDER BY [Unit Price] ASC</code>	Return cheapest products that exclude the words box, bag or carton.

Marketing Questions Part 2

The marketing team have got back to you:

- Since the winner of the social media campaign might be a child, they don't want to send a knife.
- Instead they have suggested that we send a mug or shirt to the chosen winner.
- The product should also be black if possible.

Find a list of products that contain mug or shirt in their name. How many are there?	Easy
Since it's Halloween, how many products meet this description, and are also black?	Medium
What is the WWI Stock Item ID of the cheapest product meeting the above conditions? If multiple products have the same price, choose the one with the lowest WWI Stock Item ID.	Hard
What is the markup of WWI Stock Item ID 29? $\text{markup} = (\text{retail price} - \text{unit price}) / \text{unit price}$	Hard

Black Products Containing Mug or Shirt

SQL Query	
<pre>SELECT [Stock Item] AS ProductName, [Unit Price] FROM dimStockItem WHERE ([Stock Item] LIKE '%mug%' OR [Stock Item] LIKE '%shirt%') AND Color = 'Black'</pre>	<p>Returns all mug or shirt products that are black.</p> <p>Notice the placement of the brackets in the WHERE condition.</p>
<pre>SELECT COUNT([WWI Stock Item ID]) FROM dimStockItem WHERE ([Stock Item] LIKE '%mug%' OR [Stock Item] LIKE '%shirt%') AND Color = 'Black'</pre>	<p>Returns count of all mug or shirt products that are black.</p> <p>Notice the placement of the brackets in the WHERE condition.</p>
<pre>SELECT [WWI Stock Item ID], [Stock Item] AS ProductName, [Unit Price] FROM dimStockItem WHERE ([Stock Item] LIKE '%mug%' OR [Stock Item] LIKE '%shirt%') AND Color = 'Black' ORDER BY [Unit Price] ASC</pre>	<p>Returns list of products that meet these conditions in ascending order.</p>

Mark Up

SQL Query	
<pre>SELECT [Stock Item], [Unit Price], [Recommended Retail Price], ([Recommended Retail Price] - [Unit Price]) / [Unit Price] AS PctMarkup FROM dimStockItem WHERE [WWI Stock Item ID] = 29</pre>	Return the markup of item with WWI Stock item ID 29
<pre>SELECT [Stock Item], [Unit Price], [Recommended Retail Price], CAST(([Recommended Retail Price] - [Unit Price]) / [Unit Price] as decimal(8,4)) AS PctMarkup FROM dimStockItem WHERE [WWI Stock Item ID] = 29</pre>	As above, but rounded to two decimal places.

Delivery Efficiency

The delivery team is exploring ideas to improve delivery efficiency. Instead of making deliveries to individual customer stores, the team wants to group deliveries by postcode and buying group. Buying groups purchase inventory on behalf of the stores within the group.

How many customers are in each buying group?

Easy

We are trialling a new delivery process with Wingtip Toys. We need to identify clusters of shops from this buying group that are near each other.

Do any postcodes have more than 3 Wingtip Toys shops? If so, which postcode?

Hard

If a postcode has been identified, which of the following stores should be included in the delivery efficiency trial?

Hardest

Customers by Buying Group

SQL Query	
<pre>SELECT [Buying Group], COUNT([Customer Key]) AS CustomerCount FROM dimCustomer GROUP BY [Buying Group] ORDER BY CustomerCount ASC</pre>	Returns the count of customers in each buying group.
<pre>SELECT [Postal Code] FROM dimCustomer WHERE [Buying Group] = 'Wingtip Toys' GROUP BY [Postal Code] HAVING COUNT([Customer Key]) >3</pre>	Returns a list of postcodes with more than 3 Wingtip Toy shops.

List Customers in Same Postal Code

SQL Query	
<pre>SELECT [Postal Code], Customer FROM dimCustomer WHERE [Postal Code] IN (SELECT [Postal Code] FROM dimCustomer WHERE [Buying Group] = 'Wingtip Toys' GROUP BY [Postal Code] HAVING COUNT([Customer Key]) >3)</pre>	Returns list of customers in post codes that have more than three Wingtip Customer shops.
<pre>SELECT [Postal Code], Customer FROM dimCustomer WHERE [Buying Group] = 'Wingtip Toys' AND [Postal Code] IN (SELECT [Postal Code] FROM dimCustomer WHERE [Buying Group] = 'Wingtip Toys' GROUP BY [Postal Code] HAVING COUNT([Customer Key]) >3)</pre>	As above, but filters the list to return only WingTip shops.

Other Questions

You have a little spare time and decide to explore some other dimensions in the warehouse.

Employee Dimension. What proportion of our workforce works in sales? Hard

City Dimension. Which sales territory has the highest population? Easy

City Dimension. How many cities are in the above territory? Easy

City Dimension. What is the approx. population of the biggest city in that territory? Hard

City Dimension. What is the total population across all sales territories? Hard

Make it more advanced – try to answer the city dimension questions with a single query. Hardest

Employee Dimension Queries

SQL Query	
<pre>SELECT CAST(COUNT(Employee) AS decimal(8,2)) / (SELECT COUNT(Employee) As EmployeeCnt FROM dimEmployee WHERE Employee != 'Unknown') As SalesPeoplePctOfTotal FROM dimEmployee WHERE Employee != 'Unknown' AND [Is Salesperson] = 1</pre>	% of employees working in sales
<pre>SELECT CAST(CAST(COUNT(Employee) AS decimal(8,2)) / (SELECT COUNT(Employee) As EmployeeCnt FROM dimEmployee WHERE Employee != 'Unknown') AS decimal(8,2)) As SalesPeoplePctOfTotal FROM dimEmployee WHERE Employee != 'Unknown' AND [Is Salesperson] = 1</pre>	As above, but returned to two decimal places.

City Dimension Queries

SQL Query	
<pre>SELECT [Sales Territory], SUM([Latest Recorded Population]) AS TotalPopulation, COUNT([WWI City ID]) AS NumberOfCities, MAX([Latest Recorded Population]) AS PopulationInBiggestCity FROM dimCity WHERE City != 'Unknown' GROUP BY ROLLUP ([Sales Territory]) ORDER BY TotalPopulation DESC</pre>	Returns total population, count of cities, max city population for each sales territory. Also adds a total row.
<pre>SELECT ISNULL([Sales Territory], 'Total') AS SalesTerritory, SUM([Latest Recorded Population]) AS TotalPopulation, COUNT([WWI City ID]) AS NumberOfCities, MAX([Latest Recorded Population]) AS PopulationInBiggestCity FROM dimCity WHERE City != 'Unknown' GROUP BY ROLLUP ([Sales Territory]) ORDER BY TotalPopulation DESC</pre>	As above, and tidies up the presentation of the total row.



Advanced Queries

Advanced Queries - Concepts to Recap

The questions in this section require you to pull data from a multiple tables. You may wish to review the following material from SQL Fundamentals.

Types of JOIN – [Link to Chapter From SQL Fundamentals](#)

A Basic INNER JOIN– [Link to lesson from SQL Fundamentals](#)

RIGHT JOIN to retrieve full product catalogue – [Link to lesson from SQL Fundamentals](#)

Left vs Right Join– [Link to lesson from SQL Fundamentals](#)

Investigating Sales Over Time

You have a meeting coming up with a new member of the finance team. You know they'll be interested in Sales and Profits over time. In preparation, you're going to investigate factSales and dimDate.

What granularity is the factSale table?

Easy

From the ERD, what type of relationship exists between the factSale and dimDate tables?

Easy

What is the maximum fiscal year in dimDate?

Easy

How many fiscal years do we have sales data for?

Easy

Granularity - Fact Sales

SQL Query

```
SELECT *  
FROM factSale  
WHERE [WWI Invoice ID] IN (SELECT [WWI Invoice ID] FROM factSale  
GROUP BY [WWI Invoice ID] HAVING COUNT([WWI Invoice ID]) >1 )  
ORDER BY [WWI Invoice ID]
```

Returns all sales where multiple rows appear per invoice. Used to illustrate that the grain of the fact table is not simply one row per invoice, but one row per product invoice.

Fiscal Years in Dataset

SQL Query	
<pre>SELECT MAX([Fiscal Year]) as MaxFiscalMonth FROM dimDate</pre>	Show max fiscal year in the date dimension.
<pre>SELECT d.[Fiscal Year] AS FiscalYear, SUM(s.[Total Excluding Tax]) AS TotalSalesExcludingTax FROM factSale AS s INNER JOIN dimDate AS d ON s.[Invoice Date Key]=d.[Date] GROUP BY d.[Fiscal Year] ORDER BY d.[Fiscal Year] ASC</pre>	Calculate Total Sales Excluding tax for each fiscal year.

Sales By Fiscal Period

Your meeting with finance went well. They have a few general questions for you about high-level performance.

Calculate a report of [Sales Excluding Tax], [Profit], [Quantity Sold]. You may need to analyze data by fiscal year or month.

What were the Sales Excluding tax in fiscal year 2015?

Easy

Which fiscal year appears to have the highest profit \$?

Easy

What explanation can you offer as to why the profit in 2016 is significantly lower than 2015?

Hard

Sales by Fiscal Year

SQL Query	
<pre>SELECT d.[Fiscal Year] AS FiscalYear, SUM(s.[Total Excluding Tax]) AS TotalSalesExcludingTax, SUM(s.[Quantity]) AS QuantitySold, SUM(s.[Profit]) AS Profit FROM factSale AS s INNER JOIN dimDate AS d ON s.[Invoice Date Key]=d.[Date] GROUP BY d.[Fiscal Year] ORDER BY FiscalYear DESC</pre>	Show sales and other metrics aggregated by fiscal year
<pre>SELECT d.[Fiscal Year] AS FiscalYear, FORMAT(SUM(s.[Total Excluding Tax]),'C') AS TotalSalesExcludingTax, FORMAT(SUM(s.[Quantity]),'N') AS QuantitySold, FORMAT(SUM(s.[Profit]),'C') AS Profit FROM factSale AS s INNER JOIN dimDate AS d ON s.[Invoice Date Key]=d.[Date] GROUP BY d.[Fiscal Year] ORDER BY FiscalYear DESC</pre>	Show sales and other metrics aggregated by fiscal year, and formatted nicely.

Format Function

Returns a value formatted with the specified format.

Syntax

FORMAT(value, format)

- The FORMAT function is a very simple one. When formatting numbers, you can select between “**N**”, “**C**” or “**P**” as the formatting option.
- C - currency
- N- number
- P - percentage

```
FORMAT(SUM(s.[Total Excluding Tax]), 'C')
```


Monthly Sales

SQL Query	
<pre>SELECT d.[Fiscal Month Label] AS FiscalMonth, d.[Fiscal Year] AS FiscalYear, d.[Fiscal Month Number] AS FiscalMonthNumber, SUM(s.[Total Excluding Tax]) AS TotalSalesExcludingTax, SUM(s.[Quantity]) AS QuantitySold, SUM(s.[Profit]) AS Profit FROM factSale AS s INNER JOIN dimDate AS d ON s.[Invoice Date Key]=d.[Date] GROUP BY d.[Fiscal Year],d.[Fiscal Month Label], d.[Fiscal Month Number] ORDER BY FiscalYear DESC, FiscalMonthNumber DESC</pre>	<p>Sales ordered by month and year.</p>

Top Selling Products

Sales & Marketing want to know and understand the most valuable products in the current fiscal year 2016.

What were the total sales excluding tax in fiscal year 2016?

Easy

What was the top selling product in fiscal 2016 so far?

Medium

What was the top performing product / salesperson combination in fiscal 2016?

Medium

What proportion of total fiscal year 2016 sales do these top performances represent?

Hard

Think about how you might make your query re-usable if the current year changes.

Hardest

Total Sales in 2016

SQL Query

```
SELECT SUM(s.[Total Excluding Tax])  
FROM factSale AS s  
    INNER JOIN dimDate AS d  
    ON s.[Invoice Date Key]=d.[Date]  
WHERE d.[Fiscal Year] = 2016
```

Total sales in 2016.

Top 10 Selling in YTD 2016

SQL Query

```
SELECT TOP 10
    p.[Stock Item] AS Product,
    SUM(s.[Total Excluding Tax]) AS YTDTotalSalesExcludingTax

FROM factSale AS s
    INNER JOIN dimStockItem p
    ON s.[Stock Item Key]=p.[Stock Item Key]
    INNER JOIN dimDate AS d
    ON s.[Invoice Date Key]=d.[Date]

WHERE d.[Fiscal Year] = 2016
GROUP BY p.[Stock Item]
ORDER BY YTDTotalSalesExcludingTax DESC
```

Top 10 products in YTD 2016

Sales by Salesperson and Product

SQL Query

```
SELECT TOP 10
    e.Employee AS SalesPerson,
    p.[Stock Item] AS Product,
    SUM(s.[Total Excluding Tax]) AS YTDTotalSalesExcludingTax

FROM factSale AS s
    INNER JOIN dimStockItem p
    ON s.[Stock Item Key]=p.[Stock Item Key]
    INNER JOIN dimDate AS d
    ON s.[Invoice Date Key]=d.[Date]
    INNER JOIN dimEmployee as e
    ON s.[Salesperson Key]=e.[Employee Key]

WHERE d.[Fiscal Year] = 2016
GROUP BY p.[Stock Item], e.Employee
ORDER BY YTDTotalSalesExcludingTax DESC
```

Ordered sales by employee and product in 2016.

Sales by Salesperson and Product - Percent of Total by Sales Person.

SQL Query

```
SELECT TOP 10
    e.Employee AS SalesPerson,
    p.[Stock Item] AS Product,
    SUM(s.[Total Excluding Tax]) AS YTDTotalSalesExcludingTax,
    FORMAT(CAST(SUM(s.[Total Excluding Tax]) / (SELECT SUM(s.[Total Excluding Tax])
                                                FROM factSale AS s
                                                INNER JOIN dimDate AS d
                                                ON s.[Invoice Date Key]=d.[Date]
                                                WHERE d.[Fiscal Year] = 2016)
            as decimal (8,6)), 'P4') AS PercentOfSalesYTD

FROM factSale AS s
    INNER JOIN dimStockItem p
    ON s.[Stock Item Key]=p.[Stock Item Key]
    INNER JOIN dimDate AS d
    ON s.[Invoice Date Key]=d.[Date]
    INNER JOIN dimEmployee as e
    ON s.[Salesperson Key]=e.[Employee Key]
```

Ordered sales by
employee and
product in 2016.

Including % of total.

Sales by Salesperson and Product - Percent of Total (Using Subquery)

SQL Query	
<pre>SELECT TOP 10 e.Employee AS SalesPerson, p.[Stock Item] AS Product, SUM(s.[Total Excluding Tax]) AS YTDTotalSalesExcludingTax, FORMAT(CAST(SUM(s.[Total Excluding Tax]) / (SELECT SUM(s.[Total Excluding Tax]) FROM factSale AS s INNER JOIN dimDate AS d ON s.[Invoice Date Key]=d.[Date] WHERE d.[Fiscal Year] = (SELECT MAX([Fiscal Year])) FROM factSale AS s INNER JOIN dimDate AS d ON s.[Invoice Date Key]=d.[Date])) as decimal (8,6)), 'P4') AS PercentOfSalesYTD FROM factSale AS s INNER JOIN dimStockItem p ON s.[Stock Item Key]=p.[Stock Item Key] INNER JOIN dimDate AS d ON s.[Invoice Date Key]=d.[Date] INNER JOIN dimEmployee as e ON s.[Salesperson Key]=e.[Employee Key] WHERE d.[Fiscal Year] = (SELECT MAX([Fiscal Year]) FROM factSale AS s INNER JOIN dimDate AS d ON s.[Invoice Date Key]=d.[Date]) GROUP BY p.[Stock Item], e.Employee ORDER BY YTDTotalSalesExcludingTax DESC</pre>	<p>Ordered sales by employee and product in 2016.</p> <p>Including % of total.</p> <p>Including a dynamic subquery to filter results by the most recent fiscal year.</p>

Top Selling Chiller Products

The business recently introduced chilled products into it's product line. Produce a table showing individual chiller products, with the quantity sold to date.

How many chiller products have zero quantity sold to date?

Hard

Zero Sales to Date

SQL Query

SELECT

p.[Stock Item],
SUM(Quantity) AS QuantitySold

FROM factSale AS s
RIGHT JOIN dimStockItem p
ON s.[Stock Item Key]=p.[Stock Item Key]

WHERE p.[Is Chiller Stock]=1

GROUP BY ROLLUP(p.[Stock Item])

ORDER BY QuantitySold DESC

Shows quantity sold by stock item, only for chiller stock items.