Cochin Traders Sales Analysis

By Prateek Shrestha

Problem Statement

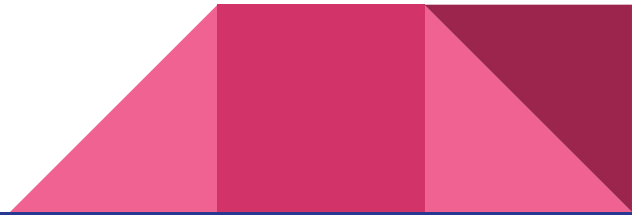
● The given Shipping database contains the sales data for a company named “Cochin Traders,” which imports and exports specialty foods from around the world.

● The company is currently facing several operational challenges, and there is a pressing need to analyze the data to identify areas for improvement in supply chain management and sales performance.

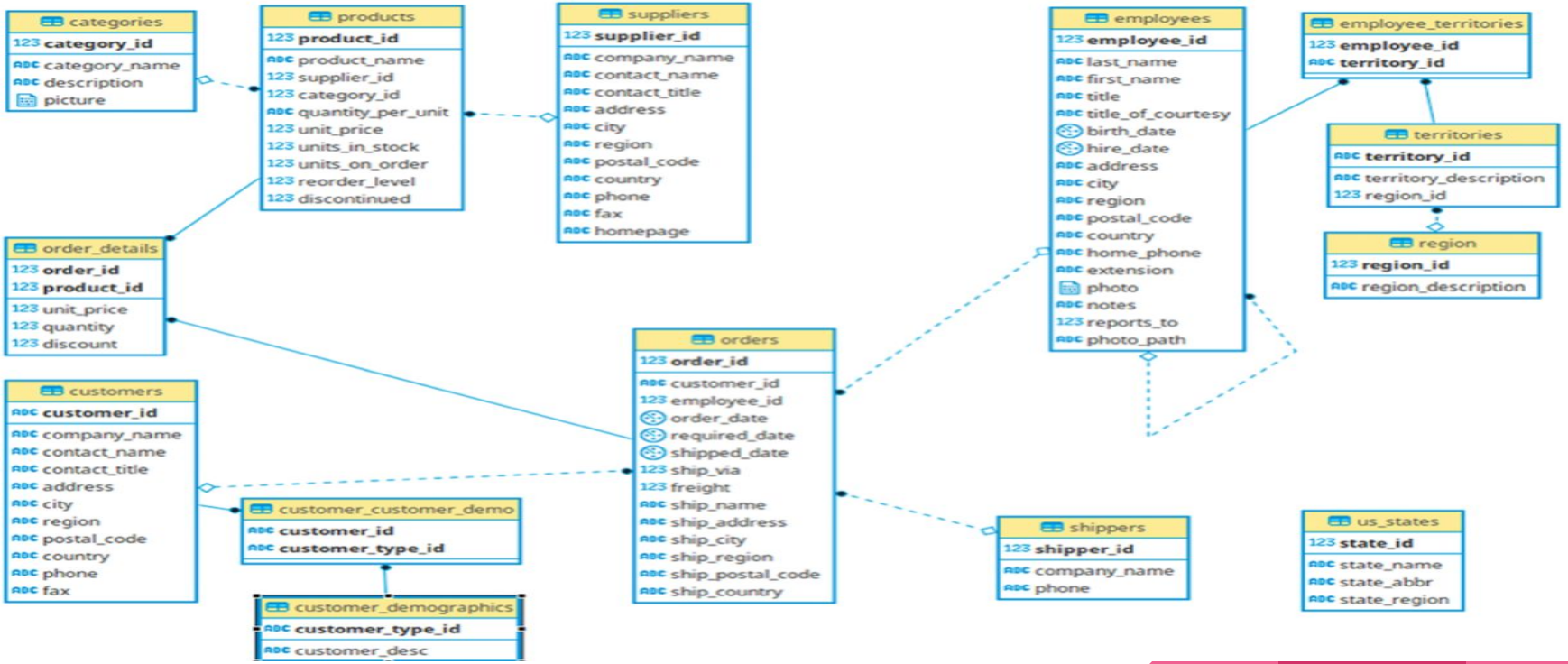
Description

The Cochin Traders database includes data for the followings:

1. **Suppliers:** Suppliers and Vendors of Cochin
2. **Customers:** Customers who buy products from Cochin
3. **Employees:** Employee details of Cochin Traders
4. **Products**: Product Information
5. **Shippers:** Details of the shippers who ship the products to the end-customers.
6. **Orders and Order_Details**: Sales order transactions taking place between the customers and the company.



Schema


categories
category_id
category_name
description
picture
products
product_id
product_name
supplier_id
category_id
quantity_per_unit
unit_price
units_in_stock
units_on_order
reorder_level
discontinued
suppliers
supplier_id
company_name
contact_name
contact_title
address
city
region
postal_code
country
phone
fax
homepage
employees
employee_id
last_name
first_name
title
title_of_courtesy
birth_date
hire_date
address
city
region
postal_code
country
home_phone
extension
photo
notes
reports_to
photo_path
employee_territories
employee_id
territory_id
territories
territory_id
territory_description
region_id
region
region_id
region_description
order_details
order_id
product_id
unit_price
quantity
discount
orders
order_id
customer_id
employee_id
order_date
required_date
shipped_date
ship_via
freight
ship_name
ship_address
ship_city
ship_region
ship_postal_code
ship_country
customers
customer_id
company_name
contact_name
contact_title
address
city
region
postal_code
country
phone
fax
customer_customer_demo
customer_id
customer_type_id
customer_demographics
customer_type_id
customer_desc
shippers
shipper_id
company_name
phone
us_states
state_id
state_name
state_abbr
state_region

1. Fetch the full names and hiring date of all Employees who work as Sales Representatives.

```sql
SELECT
  CONCAT(firstname, " ", lastname) AS Employee_name,
  hiredate
FROM `Cochin_Traders.employees`
WHERE LOWER(title) = 'sales representative'
ORDER BY Employee_name
```

Row	Employee_name	hiredate
1	Anne Dodsworth	1994-11-15
2	Janet Leverling	1992-04-01
3	Margaret Peacock	1993-05-03
4	Michael Suyama	1993-10-17
5	Nancy Davolio	1992-05-01
6	Robert King	1994-01-02

Insight: This query will provide insights into the hiring dates and tenure of employees who work as Sales Representatives at Cochin Traders.

2. Which of the products in our inventory need to be reordered?

```sql
SELECT
  productid,
  productname,
  unitsinstock,
  reorderlevel
FROM `Cochin_Traders.products`
WHERE unitsinstock <= reorderlevel
ORDER BY productid
```

Row	productid	productname	unitsinstock	reorderlevel
1	2	Chang	17	25
2	3	Aniseed Syrup	13	25
3	5	Chef Anton's Gumbo M...	0	0
4	11	Queso Cabrales	22	30
5	17	Alice Mutton	0	0
6	21	Sir Rodney's Scones	3	5
7	29	Thüringer Rostbratwurst	0	0
8	30	Nord-Ost Matjeshering	10	15
9	31	Gorgonzola Telino	0	20
10	32	Mascarpone Fabioli	9	25
11	37	Gravad lax	11	25
12	43	Ipoh Coffee	17	25
13	45	Rogede sild	5	15
14	48	Chocolade	15	25

Insights: Identifying products that need reordering ensures continuous availability, preventing stockouts and maintaining customer satisfaction.

3. An employee of ours (Margaret Peacock, EmployeeID 4) has the record of completing most orders. However, there are some customers who've never placed an order with her. Show such customers.

```
SELECT
  c.customerid,
  c.companyname
FROM `Cochin_Traders.customers` c LEFT JOIN `Cochin_Traders.orders` o
ON c.customerid = o.customerid AND o.employeeid = 4
WHERE o.employeeid IS NULL
```

Row	customerid	companyname
1	CONSH	Consolidated Holdings
2	NORTS	North/South
3	SEVES	Seven Seas Imports
4	FISSA	FISSA Fabrica Inter. Salchichas S.A.
5	DUMON	Du monde entier
6	FRANR	France restauration
7	PARIS	Paris spécialités
8	SPECD	Spécialités du monde
9	VINET	Vins et alcools Chevalier
10	PERIC	Pericles Comidas clásicas

Insights: Identifies customers who haven't placed orders with Margaret Peacock, highlighting potential gaps in her customer engagement and offering opportunities for targeted outreach or special offers.

4. The developers at Cochin Traders are testing an app that the customers will use to show orders. In order to make sure that even the largest orders will show up correctly on the app, they'd like some samples of orders that have lots of individual line items. Display the top 10 orders with the most line items.

```sql
SELECT
  orderid,
  COUNT(*) total_order_prod_count
FROM `Cochin_Traders.order_details`
GROUP BY orderid
ORDER BY total_order_prod_count DESC
LIMIT 10
```

Row	orderid	total_order_prod_count
1	11077	25
2	10657	6
3	10979	6
4	10847	6
5	10691	5
6	10325	5
7	10607	5
8	10962	5
9	10309	5
10	10382	5

Insights: This query will help the developers at Cochin Traders test the app to ensure it can handle displaying large orders with many line items correctly. By focusing on the top 10 orders with the most line items, the app's performance can be evaluated under conditions where it needs to process and display a substantial amount of data for each order.

5. Retrieve the top 5 best-selling products on the basis of the quantity ordered.

```sql
WITH top_5_prod AS (SELECT
  productid,
  SUM(quantity) AS total_quantity
FROM `Cochin_Traders.order_details`
GROUP BY productid
ORDER BY total_quantity DESC
LIMIT 5),

prod_name AS(
  SELECT
  p.productid,
  p.productname,
  t.total_quantity,
  DENSE_RANK() OVER(ORDER BY t.total_quantity DESC) AS rnk
  FROM top_5_prod t JOIN `Cochin_Traders.products` p
  ON t.productid = p.productid
)

SELECT*
FROM prod_name
ORDER BY rnk
```

Row	productid	productname	total_quantity	rnk
1	60	Camembert Pierrot	1577	1
2	59	Raclette Courdavault	1496	2
3	31	Gorgonzola Telino	1397	3
4	56	Gnocchi di nonna Alice	1263	4
5	16	Pavlova	1158	5

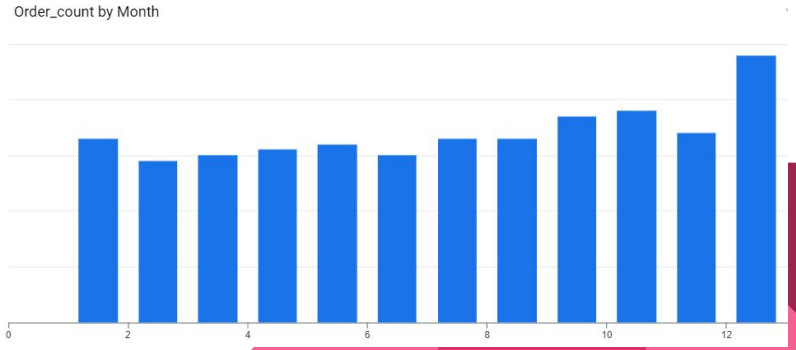
Insights: A list of the top 5 products with the highest total quantity ordered, allowing you to identify the most popular products in terms of sales volume.

6. Analyze the monthly order count for the year 1997.

```sql
WITH MonthlyOrderCount AS
(SELECT
EXTRACT(YEAR FROM orderdate) Year,
EXTRACT(MONTH FROM orderdate) Month,
FORMAT_TIMESTAMP('%b', orderdate) Month_name,
COUNT(orderid) Order_count
FROM `Cochin_Traders.orders`
GROUP BY Year, Month, Month_name
)

SELECT
  Month,
  Month_name,
  Order_count
FROM MonthlyOrderCount
WHERE Year = 1997
ORDER BY Month
```

Row	Month	Month_name	Order_count
1	1	Jan	33
2	2	Feb	29
3	3	Mar	30
4	4	Apr	31
5	5	May	32
6	6	Jun	30
7	7	Jul	33
8	8	Aug	33
9	9	Sep	37
10	10	Oct	38
11	11	Nov	34
12	12	Dec	48



Insights: Order volume remains relatively stable throughout the year, with minor fluctuations. December shows a significant peak with 48 orders. February has the lowest order count at 29.

7. Calculate the difference in sales revenue for each month compared to the previous month.

```sql
WITH monthly_revenue AS
(SELECT
  FORMAT_DATE("%Y-%m", orderdate) Year_Month,
  ROUND(SUM(od.unitprice*od.quantity),2) total_revenue
FROM `Cochin_Traders.order_details` od JOIN `Cochin_Traders.orders` o
ON od.orderid = o.orderid
GROUP BY Year_Month
ORDER BY Year_Month
)

(SELECT
  Year_Month,
  total_revenue,
  LAG(total_revenue) OVER(ORDER BY Year_Month) previous_revenu,
  ROUND((total_revenue - LAG(total_revenue) OVER(ORDER BY Year_Month)),2) revenue_diff,
  ROUND(((total_revenue - LAG(total_revenue) OVER(ORDER BY Year_Month)) / LAG(total_revenue) OVER(ORDER BY Year_Month)) * 100, 2) percentage_growth
FROM monthly_revenue
ORDER BY Year_Month)
```

Row	Year_Month	total_revenue	previous_revenu	revenue_diff	percentage_growth
1	1996-07	30192.1	*null*	*null*	*null*
2	1996-08	26609.4	30192.1	-3582.7	-11.87
3	1996-09	27636.0	26609.4	1026.6	3.86
4	1996-10	41203.6	27636.0	13567.6	49.09
5	1996-11	49704.0	41203.6	8500.4	20.63
6	1996-12	50953.4	49704.0	1249.4	2.51
7	1997-01	66692.8	50953.4	15739.4	30.89
8	1997-02	41207.2	66692.8	-25485.6	-38.21
9	1997-03	39979.9	41207.2	-1227.3	-2.98
10	1997-04	55699.39	39979.9	15719.49	39.32

Insights: provide the monthly sales revenue, the difference in sales revenue compared to the previous month, and the percentage growth in revenue, allowing you to analyze month-over-month changes more comprehensively.

8. calculate the percentage of total sales revenue for each product?

```sql
WITH TotalSales AS (
  SELECT
    ROUND(SUM(od.unitprice * od.quantity), 2) AS total_revenue
  FROM
    `Cochin_Traders.order_details` AS od
)

SELECT
  p.productname,
  ROUND(SUM(od.unitprice * od.quantity),2) AS product_revenue,
  ROUND((SUM(od.unitprice * od.quantity) / total_revenue) * 100,2) AS percentage_of_total_sales
FROM
  `Cochin_Traders.order_details` AS od JOIN `Cochin_Traders.products` AS p
ON od.productid = p.productid,
  TotalSales
GROUP BY p.productname, total_revenue
ORDER BY percentage_of_total_sales DESC;
```

Row	productname	product_revenue	percentage_of_total_sales
1	Côte de Blaye	149984.2	11.07
2	Thüringer Rostbratwurst	87736.4	6.48
3	Raclette Courdavault	76296.0	5.63
4	Camembert Pierrot	50286.0	3.71
5	Tarte au sucre	49827.9	3.68
6	Gnocchi di nonna Alice	45121.2	3.33
7	Manjimup Dried Apples	44742.6	3.3
8	Alice Mutton	35482.2	2.62
9	Carnarvon Tigers	31987.5	2.36
10	Rössle Sauerkraut	26865.6	1.98

Insights: provide each product's sales revenue and its percentage contribution to the total sales revenue, allowing you to identify the most significant products in terms of sales.