



Axon Classic Cars

SQL Case Study

Author: [Akash Katad](#)





Table Of Contents

- *Introduction*
- *Problem Statement*
- *Dataset*
- *Entity Relationship Diagram*
- *Insight's*
- *Recommendation*





Introduction

Axon Classic Cars is the go-to place for classic car enthusiasts. They specialize in sourcing, restoring, and selling iconic vehicles from automotive history, always ensuring top quality, authenticity, and customer satisfaction.

Axon needs our assistance to analyze their data for future growth.





Problem Statement

- Axon Classic Cars, a small retailer specializing in classic vehicles, struggles with sales data management and analysis.
- Their sales team lacks access to a centralized system, impeding data comprehension.
- Management faces difficulties in obtaining accurate, up-to-date sales reports, affecting decision-making.
- To resolve these issues, Axon intends to implement a Business Intelligence (BI) solution, considering PowerBI and SQL, to enhance sales data management and analysis.





Dataset

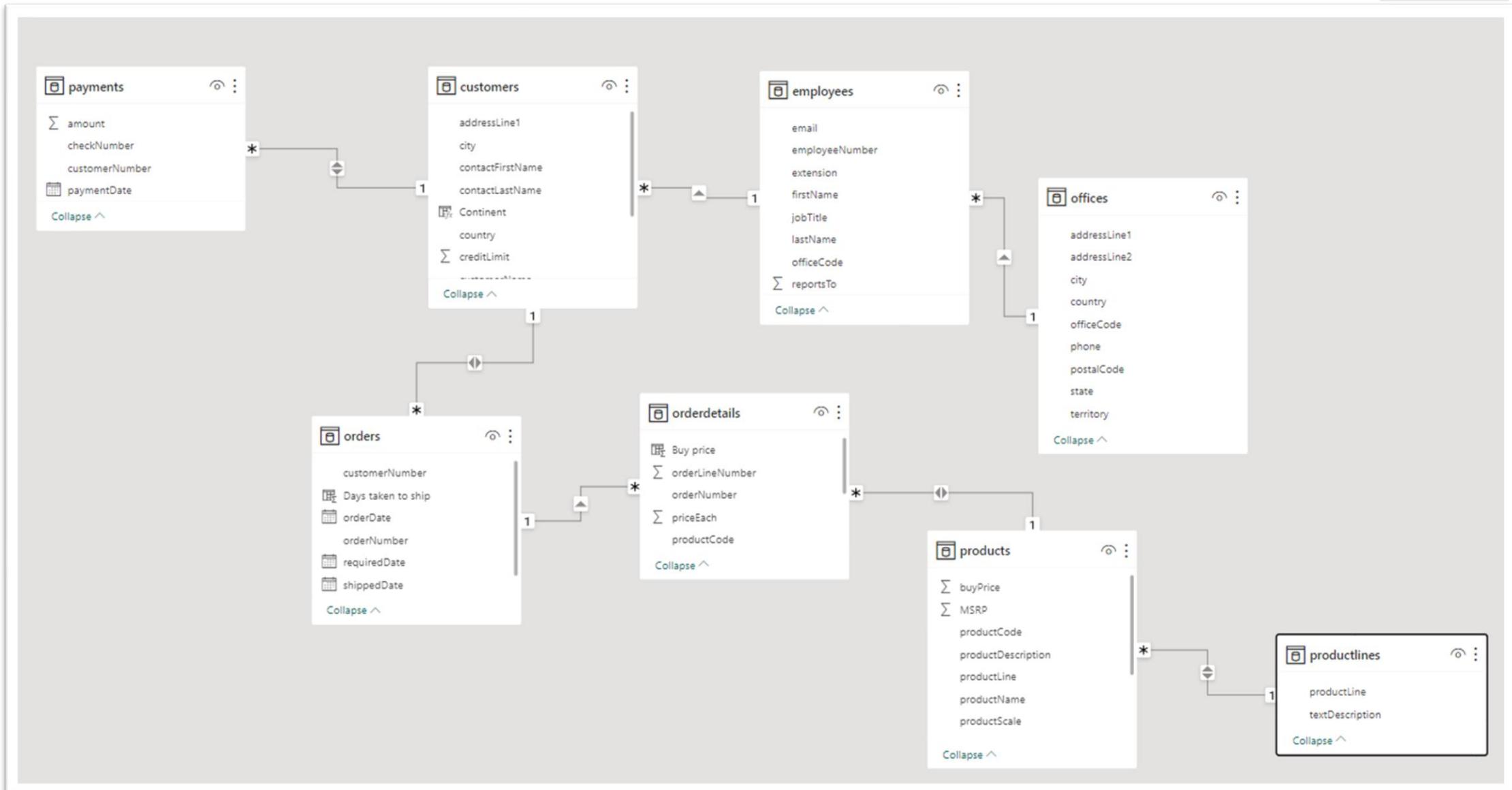
The dataset for this project is provided through a MySQL database, which comprises several key tables

- Customers:** Contains customer data, which is crucial for understanding the client base and preferences.
- Products:** Lists scale model cars available for sale.
- ProductLines:** Categorizes products into various product line categories, aiding in sales analysis.
- Orders:** Records sales orders placed by customers, offering insights into purchase trends.
- OrderDetails:** Contains detailed information about sales order line items, which is essential for a granular analysis of sales data.
- Payments:** Stores payment data, enabling tracking of customer transactions.
- Employees:** Contains information about all employees and the organization's structure, including reporting relationships.
- Offices:** Provides data on sales office locations, which may impact sales performance.





Entity Relationship Diagram





Uncovering Insights

RDBMS Used :-
MySQL



find the highest number of customers from which country

```
SELECT
    country, COUNT(*) Total_Customers
FROM
    customers
GROUP BY country
ORDER BY Total_Customers DESC
LIMIT 5;
```

country	Total_Customers
USA	36
Germany	13
France	12
Spain	7
Australia	5



find each year how many orders are placed

```
SELECT  
    YEAR(orderdate) Year, COUNT(orderdate) Total_orders  
FROM  
    orders  
GROUP BY YEAR(orderdate);
```

Year	Total_orders
2003	111
2004	151
2005	64





Write a query to find max number of orders from which month

```
SELECT  
    MONTHNAME(orderdate) Month, COUNT(*) Total_orders  
FROM  
    orders  
GROUP BY MONTHNAME(orderdate)  
ORDER BY Total_orders DESC  
LIMIT 1;
```

Month	Total_orders
November	63





find the product which is not ordered by any customer

```
SELECT  
    productName  
FROM  
    products  
WHERE  
    productCode NOT IN (SELECT  
        productCode  
    FROM  
        orderdetails);
```

productName
1985 Toyota Supra



Find The Number Of Customers For Each Sales Representative

```
SELECT
    employeenumber,
    CONCAT(firstname, ' ', lastname) FullName,
    COUNT(*) Total_customers
FROM
    customers c
        JOIN
    employees e ON c.salesrepemployeenumber = e.employeenumber
WHERE
    salesrepemployeenumber IS NOT NULL
GROUP BY 1
ORDER BY Total_customers DESC;
```

employeenumber	FullName	Total_customers
1401	Pamela Castillo	10
1504	Barry Jones	9
1323	George Vanauf	8
1501	Larry Bott	8
1286	Foon Yue Tseng	7
1370	Gerard Hernandez	7
1165	Leslie Jennings	6
1166	Leslie Thompson	6
1188	Julie Firrelli	6
1216	Steve Patterson	6
1337	Loui Bondur	6
1702	Martin Gerard	6
1611	Andy Fixter	5
1612	Peter Marsh	5
1621	Mami Nishi	5

Show customers grouped by credit limit status, divided into three categories. and show the count of customers in each group

```
CREATE VIEW Customer_Credit_Status AS
  (SELECT
    customerNumber, creditlimit,
    CONCAT(contactFirstName, contactLastName) AS Full_Name,
    CASE
      WHEN creditLimit < 10000 THEN 'Low Credit Limit'
      WHEN
        creditLimit > 10000 AND creditLimit < 75000
      THEN
        'Medium Credit Limit'
      WHEN creditLimit > 75000 THEN 'High Credit Limit'
    END AS Customer_Credit_Status
  FROM customers);
```

```
SELECT Customer_Credit_Status, COUNT(customerNumber) AS Total_Customers
FROM
  Customer_Credit_Status
GROUP BY Customer_Credit_Status
ORDER BY Total_Customers DESC;
```

	Customer_Credit_Status	Total_Customers
▶	High Credit Limit	62
	Medium Credit Limit	36
	Low Credit Limit	24



find product-wise how much stock remained in the classic model's warehouse

```
with cte as
(
  select p.productcode,productname,quantityinstock,sum(quantityordered) QuantityOrdered
  from products p join orderdetails od
  on p.productcode=od.productcode
  group by p.productcode,productname,quantityinstock
  order by QuantityOrdered desc
)

SELECT
  productname,
  (quantityinstock - (quantityordered)) AS Balance_Stock
FROM
  cte;
```

	productname	Balance_Stock
▶	1992 Ferrari 360 Spider red	6539
	1937 Lincoln Berline	7582
	American Airlines: MD-11S	7735
	1941 Chevrolet Special Deluxe Cabriolet	1302
	1930 Buick Marquette Phaeton	5988
	1940s Ford truck	2067
	1969 Harley Davidson Ultimate Chopper	6876

find the profit margin for each Product Line

```

SELECT
    productLine,
    SUM(quantityInStock) AS Quantity_in_Stock,
    SUM(quantityOrdered) AS Quantity_Orderd,
    ((SUM(quantityInStock)) - (SUM(quantityOrdered))) AS Quantity_in_Balance,
    AVG(buyPrice) avg_buy_price,
    AVG(priceEach) AS avg_selling_price,
    SUM(quantityOrdered * priceEach) AS Total_Sales,
    SUM(quantityOrdered * buyPrice) AS Total_Cost,
    (SUM(quantityOrdered * priceEach) - SUM(quantityOrdered * buyPrice)) AS Total_Profit,
    CONCAT(ROUND(((SUM(quantityOrdered * priceEach) - SUM(quantityOrdered * buyPrice)) / SUM(quantityOrdered * priceEach)) * 100,
                2),
           '%') AS Profit_Margin
FROM
    products p
        INNER JOIN
    orderdetails od ON od.productCode = p.productCode
GROUP BY productLine
ORDER BY Profit_Margin DESC;

```

productLine	Quantity_in_Stock	Quantity_Orderd	Quantity_in_Balance	avg_buy_price	avg_selling_price	Total_Sales	Total_Cost	Total_Profit	Profit_Margin
Motorcycles	1915517	12778	1902739	50.849554	87.322925	1121426.12	652170.82	469255.30	41.84 %
Vintage Cars	3439570	22933	3416637	46.017686	78.356088	1797559.63	1060291.30	737268.33	41.01 %
Classic Cars	5844033	35582	5808451	65.271901	108.004475	3853922.49	2327710.29	1526212.20	39.60 %
Ships	732251	8532	723719	46.997265	77.752041	663998.34	402708.87	261289.47	39.35 %
Trucks and Buses	1003828	11001	992827	56.329091	92.709253	1024113.57	623560.35	400553.22	39.11 %
Planes	1744036	11872	1732164	49.629167	80.327202	954637.54	588676.83	365960.71	38.34 %
-	-----	-----	-----	-----	-----	-----	-----	-----	-----



find the profit margin for each product

```

SELECT
    od.productCode,
    p.productName,
    SUM(od.quantityOrdered) AS Total_Quanitity_Orderd,
    AVG(p.buyPrice) AS Avg_Buy_Pice,
    AVG(od.priceEach) AS Avg_selling_price,
    SUM(quantityOrdered * priceEach) AS Total_sales,
    SUM(quantityOrdered * buyPrice) AS Total_Cost,
    (SUM(quantityOrdered * priceEach) - SUM(quantityOrdered * buyPrice)) AS Total_Profit,
    CONCAT(ROUND(((SUM(quantityOrdered * priceEach) -
        SUM(quantityOrdered * buyPrice)) / SUM(quantityOrdered * priceEach)) * 100,
        2),
        ' %') AS Profit_Margin
FROM
    orderdetails od
    INNER JOIN
    products p ON od.productCode = p.productCode
GROUP BY od.productCode , p.productName
ORDER BY Profit_Margin DESC;

```

productCode	productName	Avg_Buy_Pice	Avg_selling_price	Total_sales	Total_Cost	Total_Profit	Profit_Margin
S24_4620	1961 Chevrolet Impala	32.330000	73.265556	69120.97	30422.53	38698.44	55.99 %
S24_3420	1937 Horch 930V Limousine	26.300000	60.068571	52803.75	23249.20	29554.55	55.97 %
S18_2432	1926 Ford Fire Engine	24.920000	56.125714	55835.30	24870.16	30965.14	55.46 %
S12_3990	1970 Plymouth Hemi Cuda	31.920000	70.785926	63489.95	28728.00	34761.95	54.75 %
S18_2625	1936 Harley Davidson El Knucklehead	24.230000	52.609643	49992.72	22897.35	27095.37	54.20 %
S50_4713	2002 Yamaha YZR M1	34.170000	74.369259	73670.64	33896.64	39774.00	53.99 %



Provide the percentage of orders categorized by their delivery status.

```
SELECT
    status,
    CONCAT(ROUND((COUNT(DISTINCT orderNumber) * 100.0) / (SELECT
        COUNT(DISTINCT orderNumber)
    FROM
        orders),
    2),
    ' %') AS Percentage
FROM
    orders
GROUP BY status
ORDER BY Percentage DESC;
```

status	Percentage
Shipped	92.94 %
Cancelled	1.84 %
In Process	1.84 %
On Hold	1.23 %
Resolved	1.23 %
Disputed	0.92 %





Give Me The Store Procedure That Takes The Product - Code As Input And Gives The Total Orders And Total Amount

```
delimiter $  
  
create procedure Order_By_Product_code(ProductCode varchar(20))  
begin  
select pd.productCode ,pd.productName, count(distinct od.orderNumber) as Total_Orders,  
round( sum(od.quantityOrdered * od.priceEach) / 1000,2) as Total_Amount  
from products pd  
inner join orderdetails od on pd.productCode = od.productCode  
inner join orders o on o.orderNumber = od.orderNumber  
where pd.productCode = ProductCode  
group by pd.productCode,pd.productName;  
end $  
delimiter ;  
  
call Order_By_Product_code('S18_3232');
```

productCode	productName	Total_Orders	Total_Amount
S18_3232	1992 Ferrari 360 Spider red	53	276.84



Find Number Of Products Ordered By Each Vendor

```
SELECT
    p.productVendor as Vendor_Name, COUNT(od.orderNumber) AS Total_Orders
FROM
    products p
        INNER JOIN
    orderdetails od ON od.productCode = p.productCode
GROUP BY productVendor
ORDER BY Total_Orders DESC;
```

Vendor_Name	Total_Orders
Classic Metal Creations	270
Motor City Art Classics	249
Carousel DieCast Legends	246
Unimax Art Galleries	244
Gearbox Collectibles	242
Exoto Designs	240
Highway 66 Mini Classics	222
Autoart Studio Design	221
Min Lin Diecast	220
Second Gear Diecast	220
Studio M Art Models	217
Welly Diecast Productions	216
Red Start Diecast	189

Create a stored procedure that takes a customer's name as input and returns the following information:

- Customer Name
- Sales Representative Number
- Total Bill
- Amount Paid by the Customer
- Payment Status (Amount Paid or Pending)
- Pending Amount
- The procedure should provide details about the customer's financial transactions,
- including their payment status and pending amount.

```
delimiter $  
create procedure Cutomer_Payment_Status(Customer_Name varchar(150))  
begin  
select pg.customerNumber,ar.customerName,ar.country, ar.Sales_Representative ,  
pg.Total_Amt as Total_Bill ,  
ar.Total_Amount as Amount_Received ,  
case  
when  
    ar.Total_Amount < pg.Total_Amt then 'Payment Pending'  
    else 'All Payment Recived'  
end as Payment_Status,  
sum(pg.Total_Amt - ar.Total_Amount ) as Pending_Amount  
from Amount_Received ar  
inner join Price_Generated pg on pg.customerNumber = ar.customerNumber  
where ar.customerName = Customer_Name  
group by pg.customerNumber,ar.customerName;  
end $  
delimiter ;  
  
call Cutomer_Payment_Status('Euro+ Shopping Channel');
```

customerNumber	customerName	country	Sales_Representative	Total_Bill	Amount_Received	Payment_Status	Pending_Amount
141	Euro+ Shopping Channel	Spain	1370	820689.54	715738.98	Payment Pending	104950.56



Key Insights

- **Customer Distribution:** The majority of our customers are located in the USA, with Germany following closely.
- **Popular Product Lines:** Among our product lines, "Classic Cars" stands out as the most frequently ordered, closely followed by "Vintage Cars."
- **Top-Selling Product:** The product that has seen the highest number of orders from our customers is the "1992 Ferrari 360 Spider Red," with the "1937 Lincoln Berline" as the runner-up.
- **Total Sales:** The Company has achieved a commendable total sales figure of \$8.85 million to date.
- **Paris Office Excellence:** Our Paris office stands out in terms of workforce and sales production, surpassing other office locations.
- **Top-Performing Employee:** Gerard Hernandez from our Paris office has demonstrated exceptional promise as our most prolific employee, generating the highest number of orders.



Recommendation

- **Market Expansion:** While the USA is a significant market, consider leveraging the success in Germany as a stepping-stone to further expand into European markets. Invest in marketing and customer engagement strategies specifically tailored to European audiences to tap into this growth potential.
- **Product Line Expansion:** Capitalize on the popularity of "Classic Cars" and "Vintage Cars" by expanding these product lines. Introduce new models, variations, or related accessories to cater to the existing customer interest and potentially attract new customers with similar preferences
- **Promote Best-selling Products Focus** marketing efforts on promoting the "1992 Ferrari 360 Spider Red" and the "1937 Lincoln Berline" as top-selling products to maximize sales and customer engagement.
- **USA Market Expansion:** Prioritize and expand efforts in the USA market due to its significant contribution of 34.38% of total sales, aiming to sustain and potentially increase this market share over time.
- **Employee Recognition and Development:** Implement an employee recognition program to celebrate and reward outstanding performance. This can boost morale and motivate all employees to excel. Offer career development paths and opportunities for growth within the organization to retain top talent and allow them to advance their careers.



Thank You!

To explore the Interactive Report, visit

- [[Sales Report](#)]

For more About the Project, visit

- [[GitHub](#)]

Author: [Akash Katad](#)

