Customers’ Orders Analysis Using MySQL

We have analysed the **classicmodels** dataset providing insights into customer behavior, product performance, and sales trends.

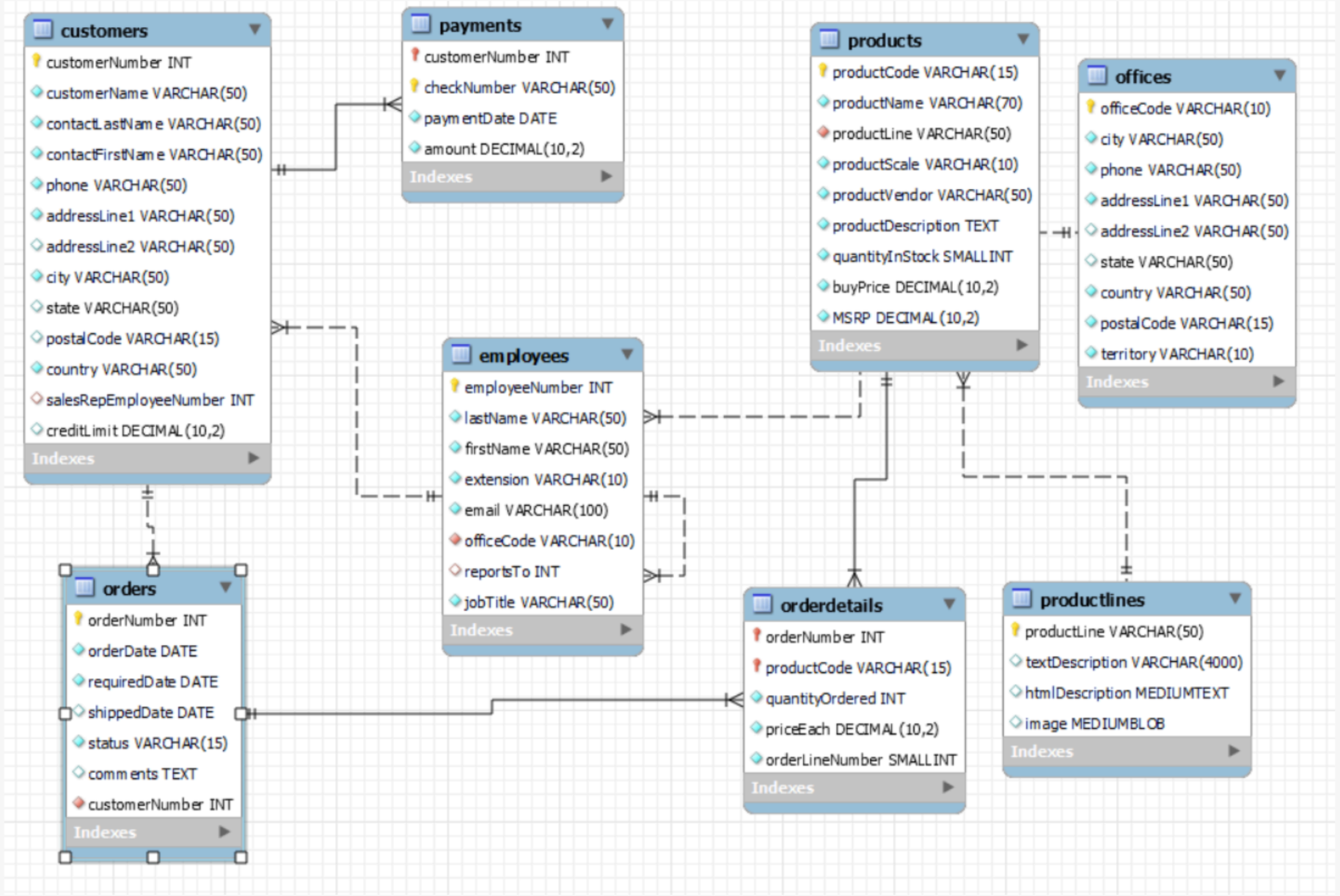
Content

The **classicmodels** database simulates a company that sells scale models of classic cars. It contains information about customers, products, product lines, orders, order details, payments, employees, and sales offices

Structure

The data is organized into eight relational tables, each representing a specific entity in the business model. These tables are linked together using foreign keys, allowing to perform complex queries across the dataset.

Database Schema



Total Customers from each country

Query

```
select country,count(customerNumber) total_customers from customers
group by country order by 2 desc;
```

Output

country	total_customers
USA	36
Germany	13
France	12
Spain	7
Australia	5
UK	5
Italy	4
New Zealand	4
Norway	3
Singapore	3
Finland	3
Canada	3
Switzerland	3

Note: The whole table is quite long so I have displayed a part of it which will give the idea about how the table looks like. I have also sorted it in descending order.

How many orders has each customer placed?

Query

```
select o.customerNumber,
(select customerName from customers c
where c.customerNumber=o.customerNumber)
customer_name,
count(orderNumber) total_orders
from orders o
where customerNumber in
(select customerNumber from customers c)
group by o.customerNumber order by 3 desc;
```

Output

customerNumber	customer_name	total_orders
141	Euro+ Shopping Channel	26
124	Mini Gifts Distributors Ltd.	17
114	Australian Collectors, Co.	5
353	Reims Collectables	5
145	Danish Wholesale Imports	5
148	Dragon Souveniers, Ltd.	5
323	Down Under Souveniers, Inc	5
381	Royale Belge	4
276	Anna's Decorations, Ltd	4
119	La Rochelle Gifts	4
121	Baane Mini Imports	4
128	Blauer See Auto, Co.	4
131	Land of Toys Inc.	4

Note: The whole table is quite long so I have displayed a part of the table which will give the idea about how the table looks like. I have also sorted it in descending order.

What is the total amount spent by each customer?

Query

```
select c.customerNumber,c.customerName,
sum(ifnull(p.amount,0)) amount_spent
from customers c
left join payments p
on c.customerNumber=p.customerNumber
group by c.customerNumber order by 3 desc;
```

Output

customerNumber	customerName	amount_spent
141	Euro + Shopping Channel	715738.98
124	Mini Gifts Distributors Ltd.	584188.24
114	Australian Collectors, Co.	180585.07
151	Muscle Machine Inc	177913.95
148	Dragon Souveniers, Ltd.	156251.03
323	Down Under Souveniers, Inc	154622.08
187	AV Stores, Co.	148410.09
276	Anna's Decorations, Ltd	137034.22
321	Corporate Gift Ideas Co.	132340.78
146	Saveley & Henriot, Co.	130305.35
278	Rovelli Gifts	127529.69
353	Reims Collectables	126983.19
119	La Rochelle Gifts	116949.68

Note: The whole table is quite long so I have displayed a part of the table which will give the idea about how the table looks like. I have also sorted it in descending order.

List the top 5 most expensive products(average price across all orders).

Query

```
select od.productCode,
(select productName from products
p where p.productCode=od.productCode) product_name,
avg(od.priceEach) average_price
from orderdetails od
group by od.productCode,product_name
order by average_price desc limit 5;
```

Output

productCode	product_name	average_price
S10_1949	1952 Alpine Renault 1300	197.309286
S12_1108	2001 Ferrari Enzo	187.096667
S12_1099	1968 Ford Mustang	172.446296
S10_4698	2003 Harley-Davidson Eagle Drag Bike	172.288214
S12_3891	1969 Ford Falcon	158.218889

Display salesperson and the number of sales made?

Query

```
select salesRepEmployeeNumber,
(select concat(firstname,' ',lastname)
from employees e where c.salesRepEmployeeNumber=e.employeeNumber) Name,
count(customerNumber) total_no_of_sales
from customers c
where c.customerNumber in
(select customerNumber from orders o where o.status in ('Shipped','Resolved'))
group by salesRepEmployeeNumber order by total_no_of_sales desc;
```

Output

salesRepEmployeeNumber	Name	total_no_of_sales
1401	Pamela Castillo	10
1504	Barry Jones	9
1323	George Vanauf	8
1501	Larry Bott	8
1370	Gerard Hernandez	7
1165	Leslie Jennings	6
1166	Leslie Thompson	6
1188	Julie Firrelli	6
1216	Steve Patterson	6

What is the average order value?

Query

```
select round(avg(total_price),2) average_order_price
from (select o.orderNumber,
sum(od.quantityOrdered*od.priceEach) total_price
from orders o join orderdetails od
on o.orderNumber=od.orderNumber group by o.orderNumber) a;
```

Output

average_order_price
29460.71

Retrieve least ordered product and most ordered product.

Query

```
select p.productCode,p.productName,count(o.orderNumber) Order_count
from products p
join orderdetails o on p.productCode=o.productCode
group by p.productCode , p.productName
having Order_count in
(select min(ordercount) from
(select count(orderNumber) ordercount from orderdetails od group by od.productCode) a);
-- ----------------------------------------
select p.productCode,p.productName,count(o.orderNumber) Order_count
from products p
join orderdetails o on p.productCode=o.productCode
group by p.productCode , p.productName
having Order_count in (select max(ordercount)
from (select count(orderNumber) ordercount
from orderdetails od group by od.productCode) a);
```

Output

For most ordered

productCode	productName	Order_count
S18_3232	1992 Ferrari 360 Spider red	53

For least ordered

productCode	productName	Order_count
S18_4933	1957 Ford Thunderbird	24
S24_2887	1952 Citroen-15CV	24

List number of products from all product line.

Query

```
select productLine,count(productCode) number_of_products
from products group by productLine order by 2 desc;
```

Output

productLine	number_of_products
Classic Cars	38
Vintage Cars	24
Motorcycles	13
Planes	12
Trucks and Buses	11
Ships	9
Trains	3

Find the products that were ordered top 2 times for each country.

Query

```
select productCode,productName,Order_count, country from
(select p.productCode productCode,p.productName productName,
count(o.orderNumber) Order_count,c.country country,
dense_rank() over(partition by c.country
order by count(o.orderNumber) desc) as order_rank
from customers c join orders o
on c.customerNumber=o.customerNumber
join orderdetails od on o.orderNumber=od.orderNumber
join products p on od.productCode=p.productCode
group by p.productCode,p.productName,c.country
order by 3 desc, 4) a where a.order_rank<=2 order by country;
```

Output

productCode	productName	Order_count	country
S18_2949	1913 Ford Model T Speedster	6	Australia
S12_1666	1958 Setra Bus	5	Australia
S18_1097	1940 Ford Pickup Truck	5	Australia
S12_3380	1968 Dodge Charger	3	Austria
S12_4675	1969 Dodge Charger	3	Austria
S12_1099	1968 Ford Mustang	2	Austria
S12_3990	1970 Plymouth Hemi Cuda	2	Austria

Note: The whole table is quite long so I have displayed a part of the table which will give the idea about how the table looks like.

Calculate revenue, gross profit margin for each product across orders that are 'Shipped' and 'Resolved'.

Query

```
select p.productCode, p.productName,
sum(od.priceEach * od.quantityOrdered) revenue,
sum((od.priceEach - p.buyPrice) * od.quantityOrdered) grossProfit,
round((sum((od.priceEach - p.buyPrice) * od.quantityOrdered)*100/sum(od.priceEach * od.quantityOrdered)),2) grossProfitMargin
from products p join orderdetails od on p.productCode = od.productCode
where od.orderNumber in (select orderNumber from orders o where o.status in ('Shipped','Resolved'))
group by p.productCode, p.productName;
```

Output

productCode	productName	revenue	grossProfit	grossProfitMargin
S18_1749	1917 Grand Touring Sedan	124151.00	54357.50	43.78
S18_2248	1911 Ford Town Car	40991.57	15883.37	38.75
S18_4409	1932 Alfa Romeo 8C2300 Spider Sport	66120.64	31772.20	48.05
S24_3969	1936 Mercedes Benz 500k Roadster	29234.04	11638.29	39.81
S18_2325	1932 Model A Ford J-Coupe	104728.81	51395.05	49.07
S18_2795	1928 Mercedes-Benz SSK	126428.88	65115.68	51.50
S24_1937	1939 Chevrolet Deluxe Coupe	26604.39	6471.95	24.33
S24_2022	1938 Cadillac V-16 Presidential Limousine	36657.09	17798.94	48.56
S18_1342	1937 Lincoln Berline	97900.08	33642.88	34.36
S18_1367	1936 Mercedes-Benz 500K Special Roadster	44892.29	22209.19	49.47
S10_1949	1952 Alpine Renault 1300	179945.96	90139.58	50.09

Note: The whole table is quite long so I have displayed a part of the table which will give the idea about how the table looks like.

Show the total sales by product categories per quarter for the years.

```
select year(orderDate) year, quarter(orderDate) quarter,
(select p.productLine from products p where p.productCode=od.productCode) productLine,
sum(od.quantityOrdered*od.priceEach) total_sales
from orders o join orderdetails od
on o.orderNumber=od.orderNumber
where o.status in ('Shipped','Resolved')
group by 1,2,3 order by 1,2;
```

Query

Note: The whole table is quite long so I have displayed a part of the table which will give the idea about how the table looks like.

Output

year	quarter	productLine	total_sales
2003	1	Classic Cars	152581.55
2003	1	Motorcycles	33062.22
2003	1	Planes	37136.27
2003	1	Ships	24446.99
2003	1	Trains	7810.61
2003	1	Trucks and Buses	43593.71
2003	1	Vintage Cars	106982.20
2003	2	Classic Cars	194291.40
2003	2	Motorcycles	43490.68
2003	2	Planes	66587.81
2003	2	Ships	50171.58
2003	2	Trains	12144.96
2003	2	Trucks and Buses	62804.84

Conclusion

What we have learned

- Identified countries with the highest number of customers
- Revealed how many orders each customer has placed, highlighting the most active customers.
- Revealed which customers contribute the most to revenue.
- Listed top products with the highest average prices across all orders.
- Displayed the number of sales made by each salesperson, identifying top performers.
- Calculated the average value of orders, providing an overview of typical order size.
- Identified the least and most ordered products, giving insights into product demand.
- Listed the number of products in each product line, highlighting product line diversity.
- Revealed the most ordered products for each country, showing regional preferences.
- Indentified revenue and gross profit margins from different products across successful orders
- Analyzed total sales by product categories for each quarter, providing seasonal sales insights.