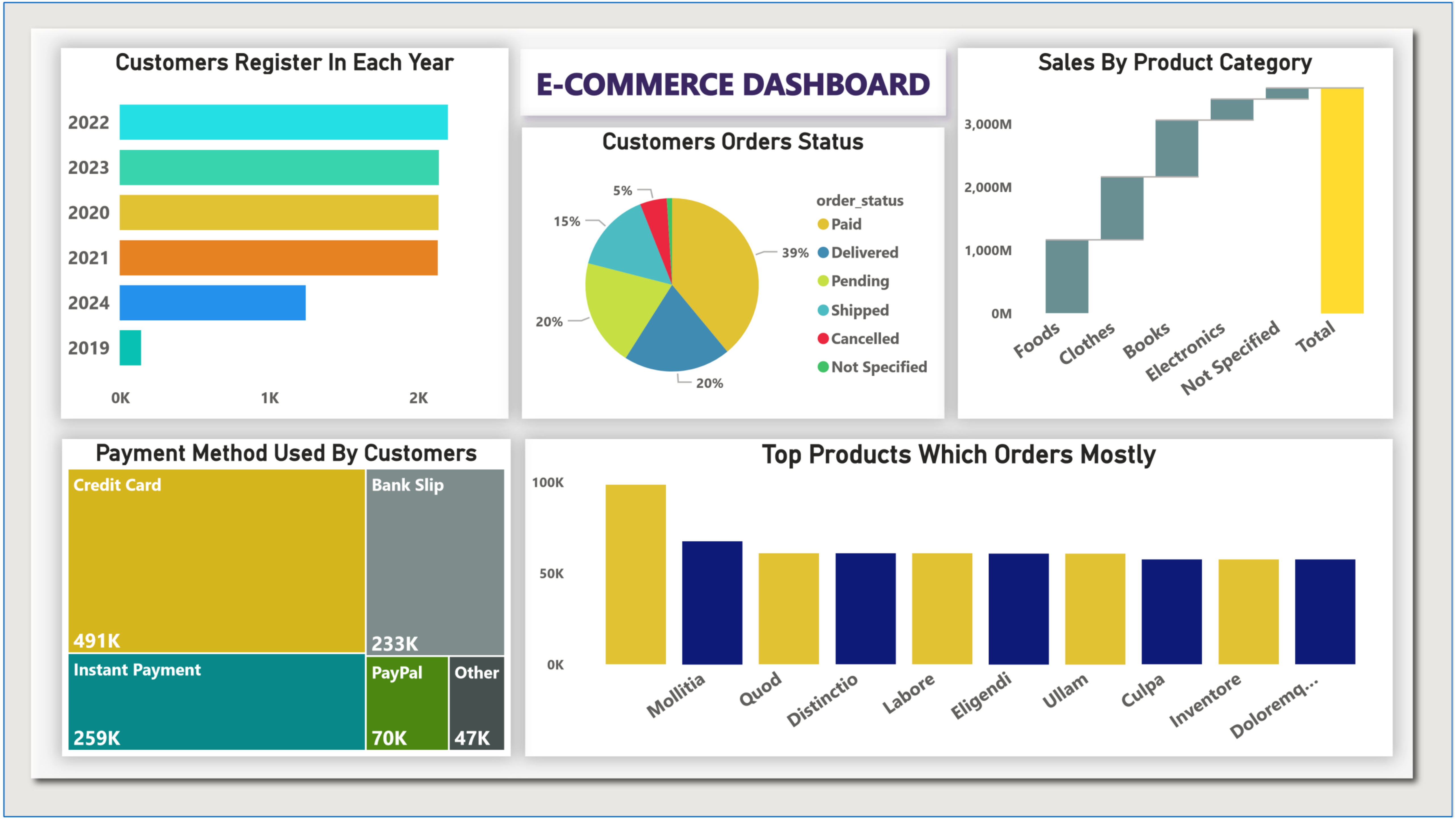


E-COMMERCE DASHBOARD

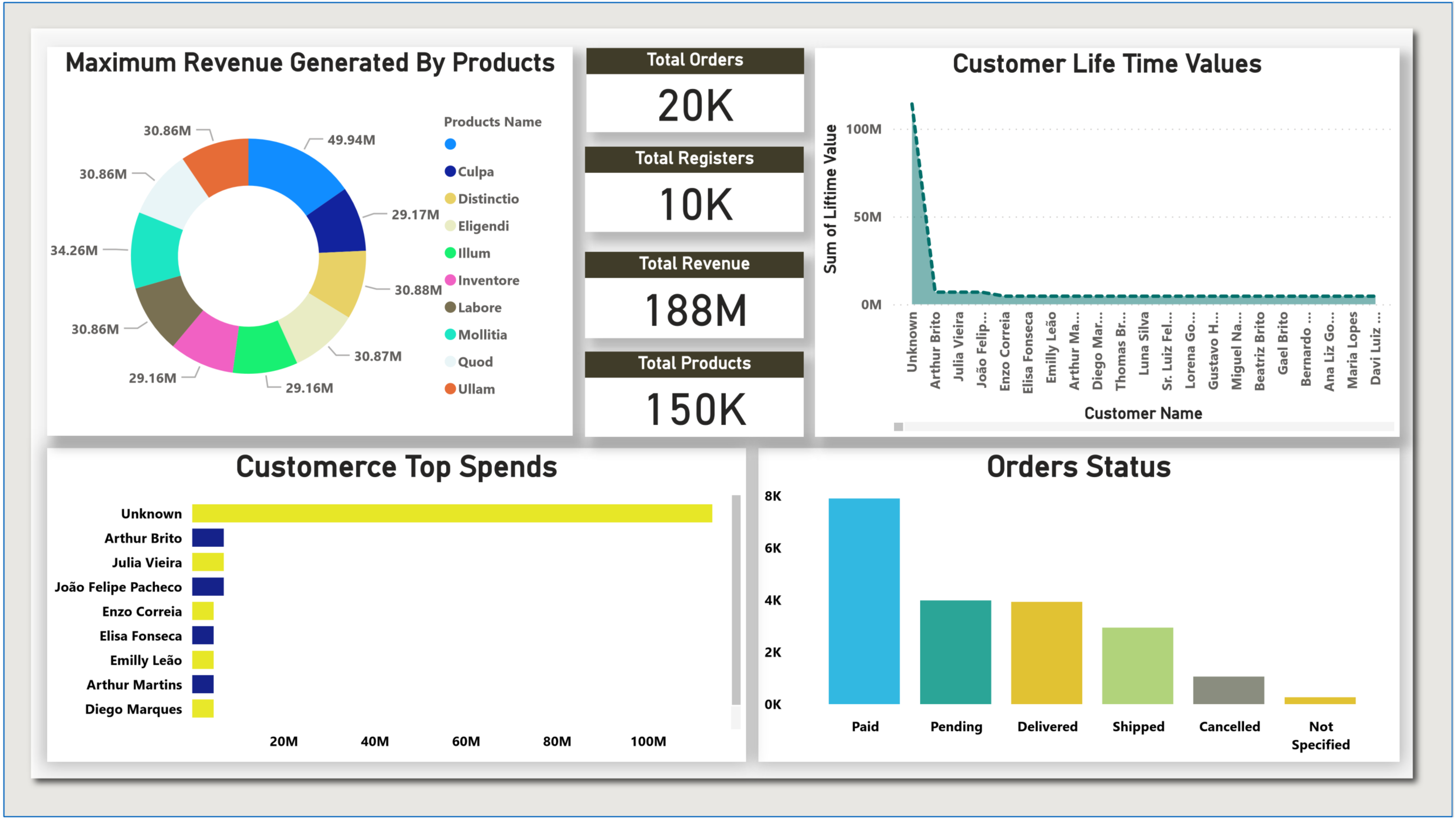
Customers Register In Each Year

Customers Orders Status

Sales By Product Category

Payment Method Used By Customers

Top Products Which Orders Mostly



Maximum Revenue Generated By Products

Total Orders
20K

Total Registers
10K

Total Revenue
188M

Total Products
150K

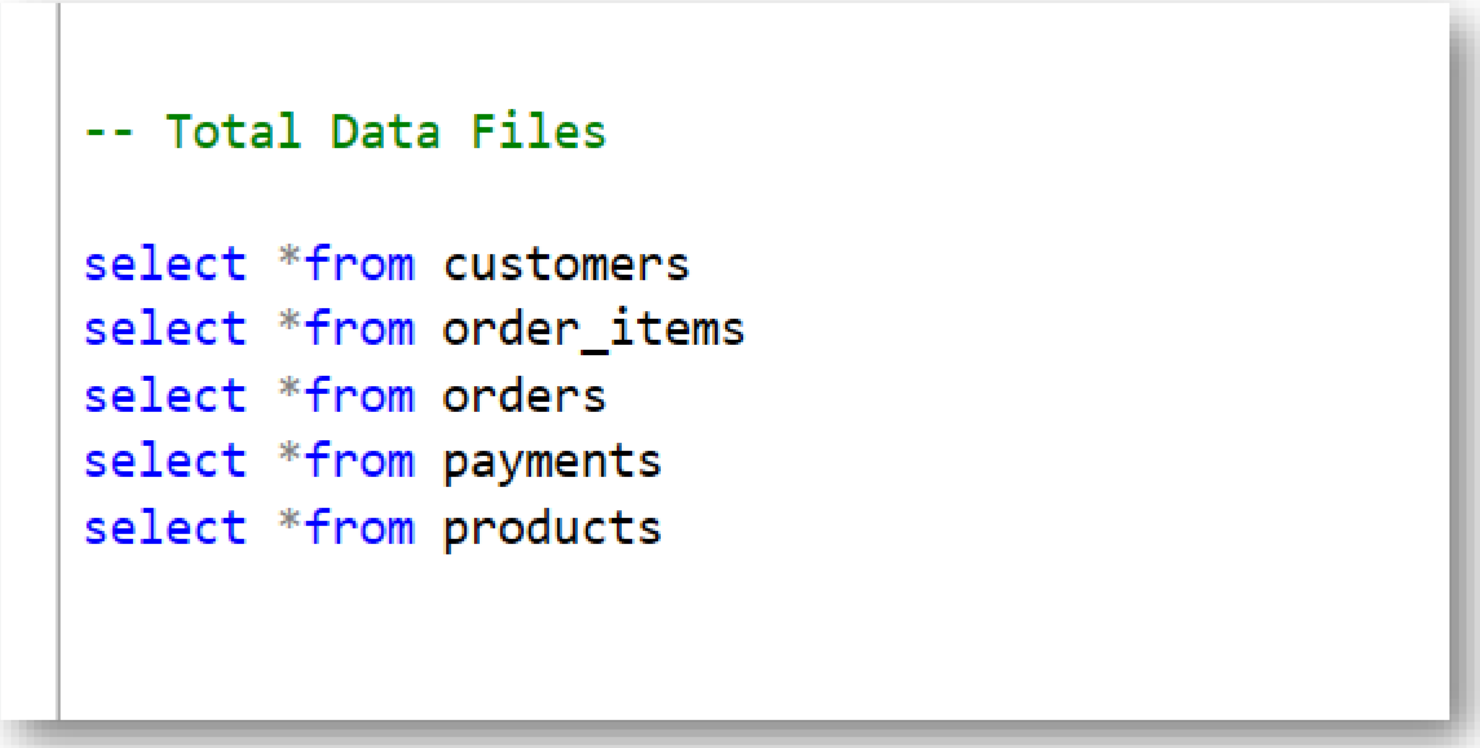
Customer Life Time Values

Customerce Top Spends

Orders Status

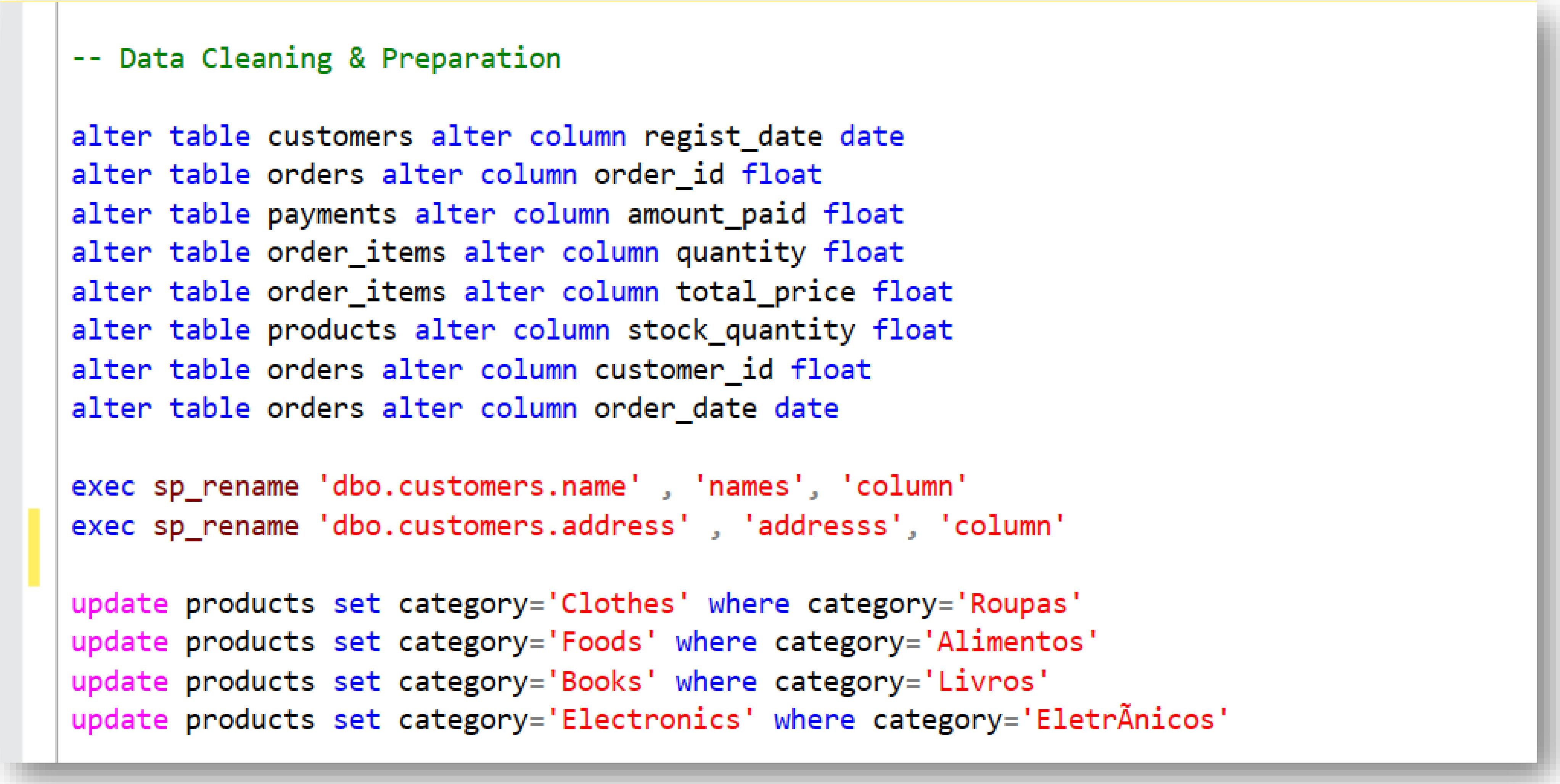
Data Cleaning &Preparation With SQL

Data Files:

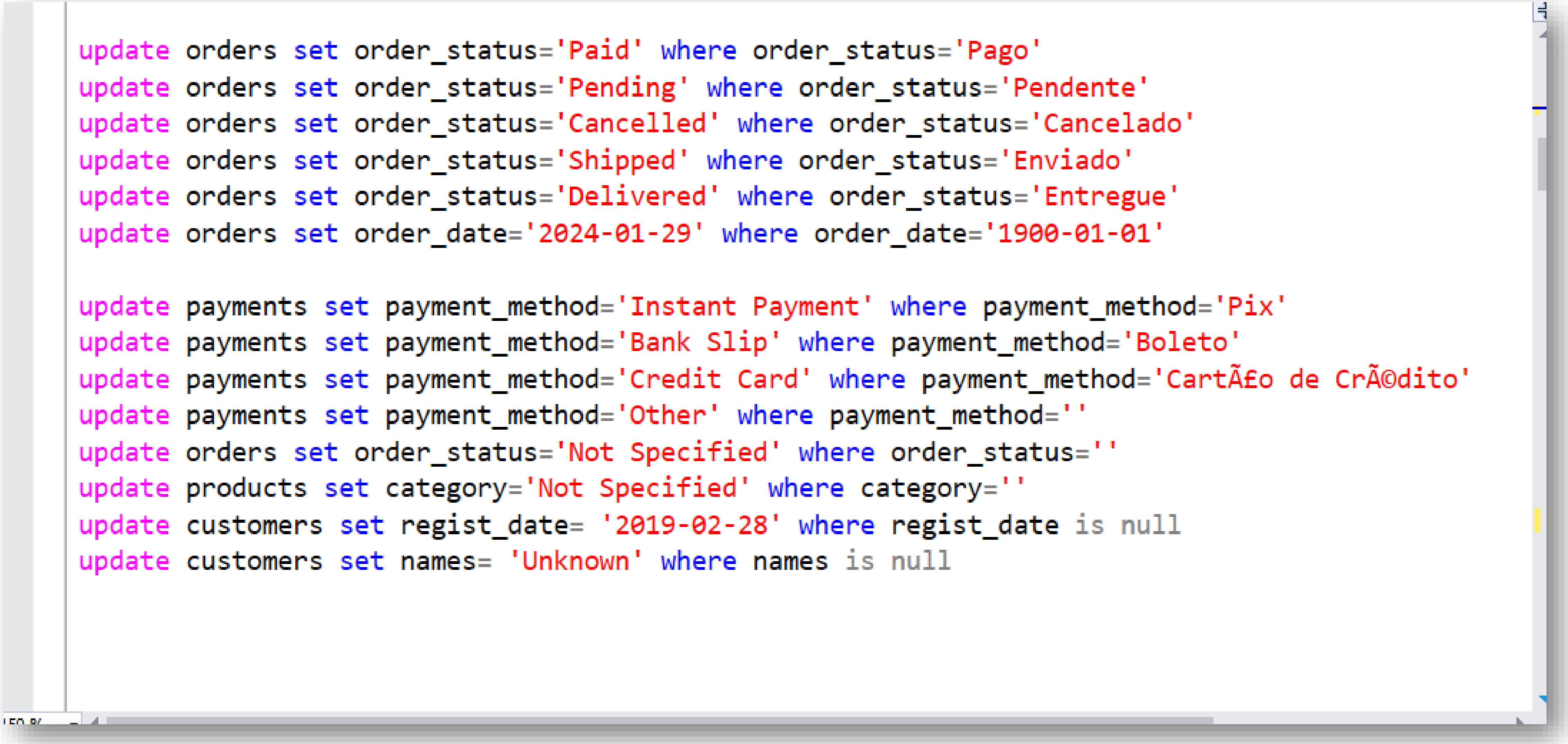


```
-- Total Data Files

select *from customers
select *from order_items
select *from orders
select *from payments
select *from products
```

Data Cleaning:



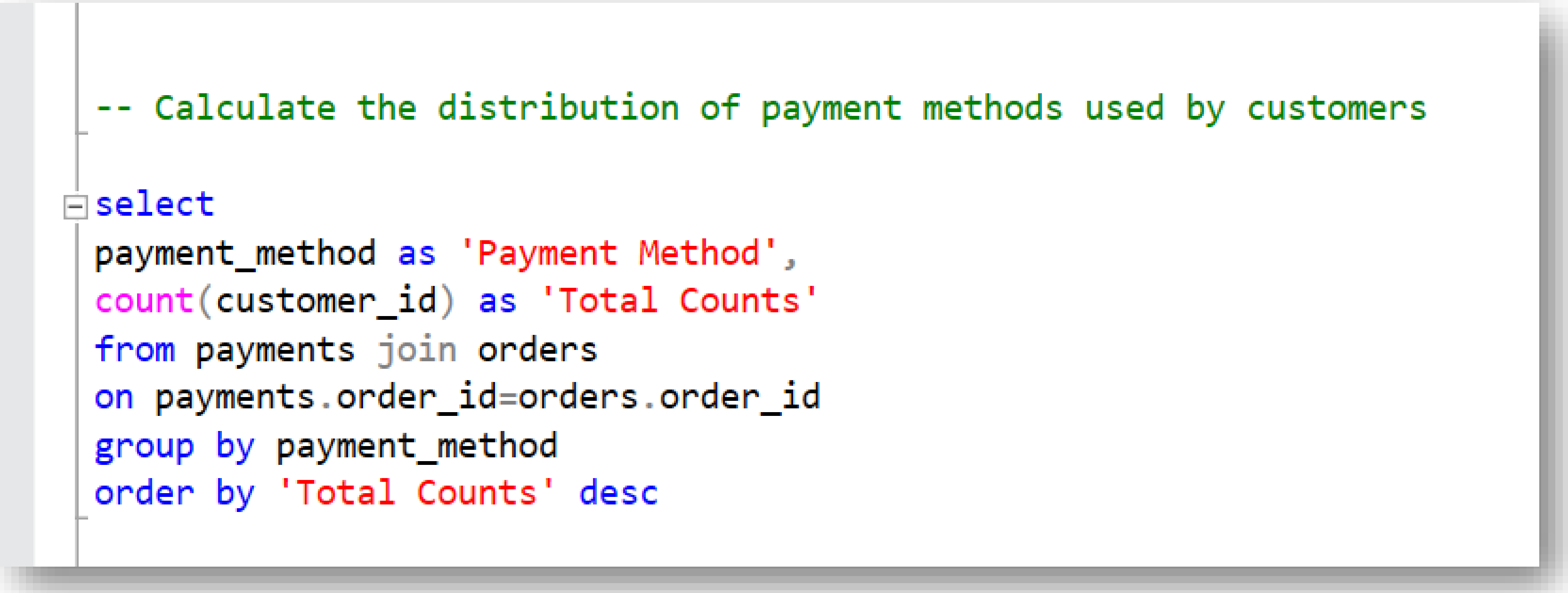
```sql
-- Data Cleaning & Preparation

alter table customers alter column regist_date date
alter table orders alter column order_id float
alter table payments alter column amount_paid float
alter table order_items alter column quantity float
alter table order_items alter column total_price float
alter table products alter column stock_quantity float
alter table orders alter column customer_id float
alter table orders alter column order_date date

exec sp_rename 'dbo.customers.name' , 'names', 'column'
exec sp_rename 'dbo.customers.address' , 'addresss', 'column'

update products set category='Clothes' where category='Roupas'
update products set category='Foods' where category='Alimentos'
update products set category='Books' where category='Livros'
update products set category='Electronics' where category='EletrÃnicos'
```



```sql
update orders set order_status='Paid' where order_status='Pago'
update orders set order_status='Pending' where order_status='Pendente'
update orders set order_status='Cancelled' where order_status='Cancelado'
update orders set order_status='Shipped' where order_status='Enviado'
update orders set order_status='Delivered' where order_status='Entregue'
update orders set order_date='2024-01-29' where order_date='1900-01-01'

update payments set payment_method='Instant Payment' where payment_method='Pix'
update payments set payment_method='Bank Slip' where payment_method='Boleto'
update payments set payment_method='Credit Card' where payment_method='CartÃ£o de CrÃ©dito'
update payments set payment_method='Other' where payment_method=''
update orders set order_status='Not Specified' where order_status=''
update products set category='Not Specified' where category=''
update customers set regist_date= '2019-02-28' where regist_date is null
update customers set names= 'Unknown' where names is null
```

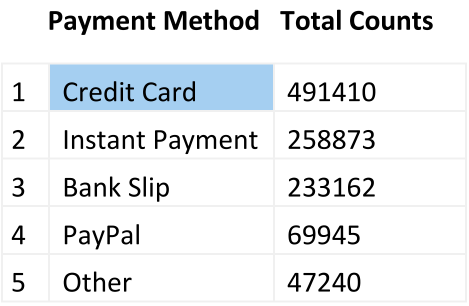
Data Analysis:

Query:



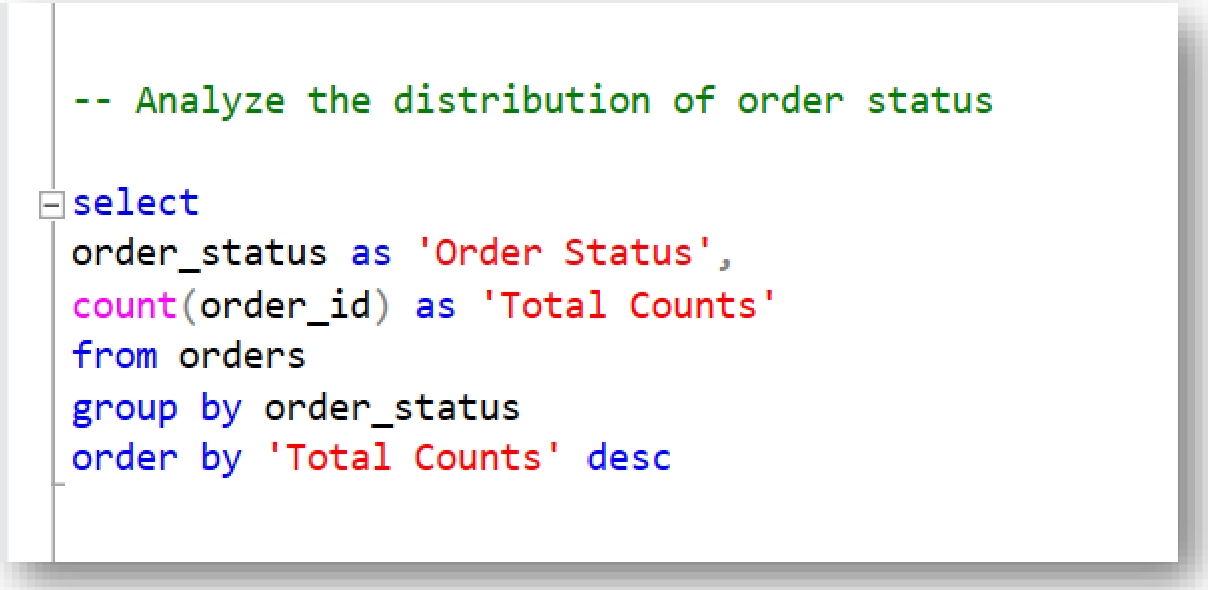
```sql
-- Calculate the distribution of payment methods used by customers

select
payment_method as 'Payment Method',
count(customer_id) as 'Total Counts'
from payments join orders
on payments.order_id=orders.order_id
group by payment_method
order by 'Total Counts' desc
```

Output:



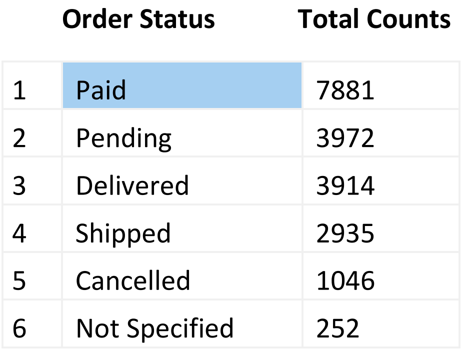
	Payment Method	Total Counts
1	Credit Card	491410
2	Instant Payment	258873
3	Bank Slip	233162
4	PayPal	69945
5	Other	47240

Query:



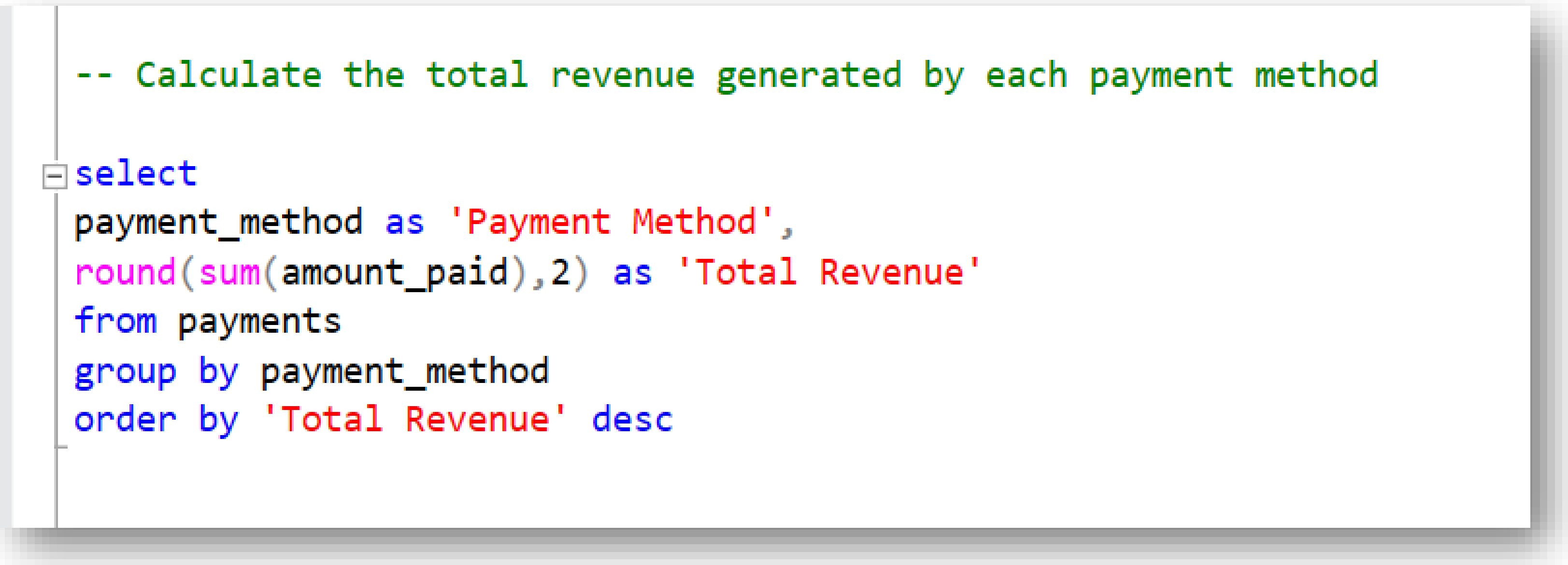
```
-- Analyze the distribution of order status

select
order_status as 'Order Status',
count(order_id) as 'Total Counts'
from orders
group by order_status
order by 'Total Counts' desc
```

Output:



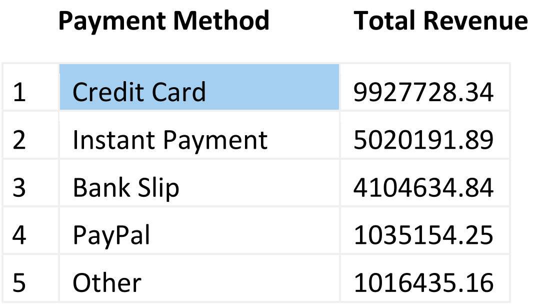
	Order Status	Total Counts
1	Paid	7881
2	Pending	3972
3	Delivered	3914
4	Shipped	2935
5	Cancelled	1046
6	Not Specified	252

Query:



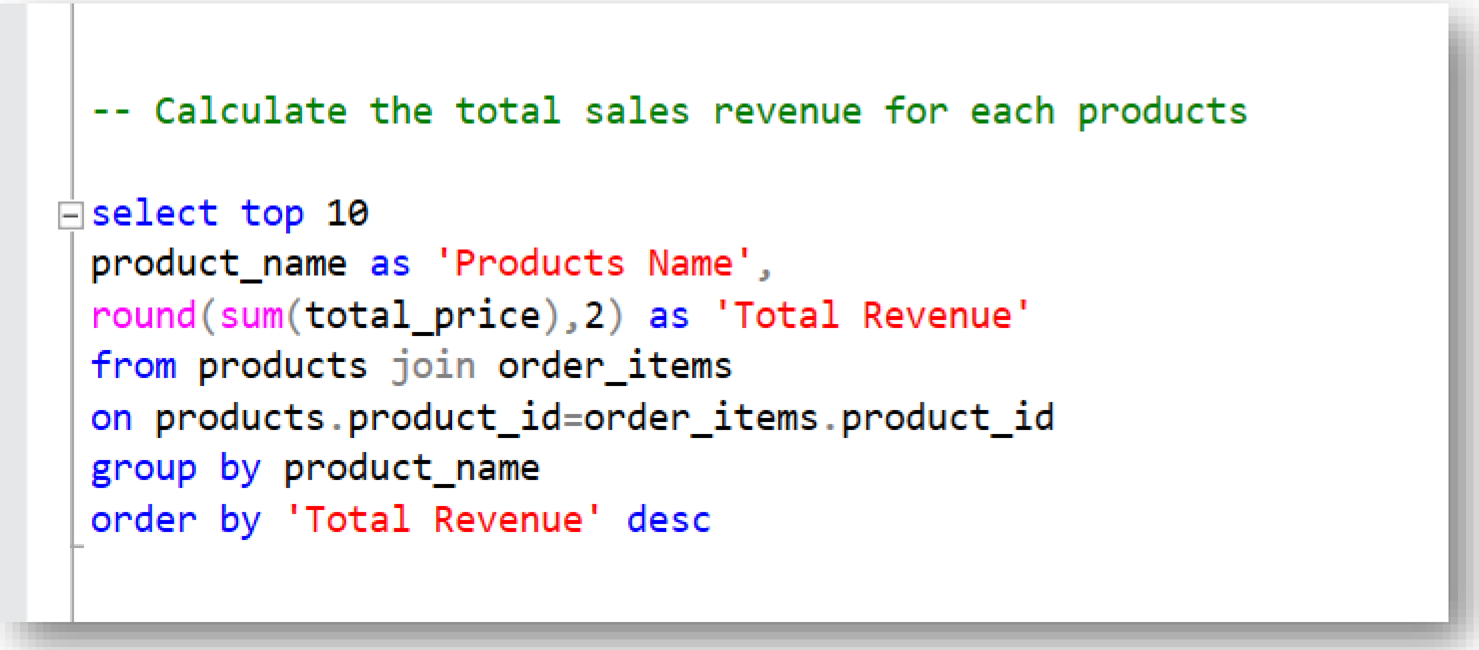
```
-- Calculate the total revenue generated by each payment method

select
payment_method as 'Payment Method',
round(sum(amount_paid),2) as 'Total Revenue'
from payments
group by payment_method
order by 'Total Revenue' desc
```

Output:



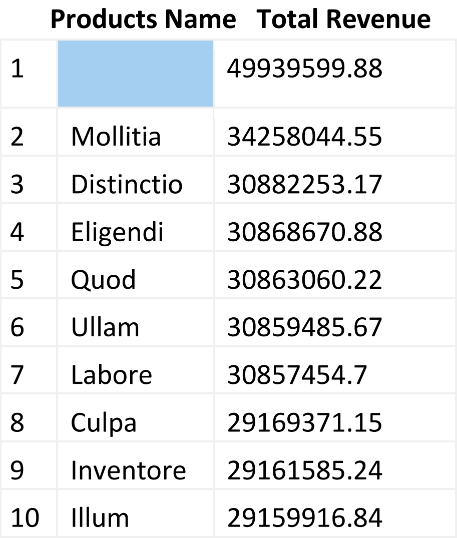
	Payment Method	Total Revenue
1	Credit Card	9927728.34
2	Instant Payment	5020191.89
3	Bank Slip	4104634.84
4	PayPal	1035154.25
5	Other	1016435.16

Query:



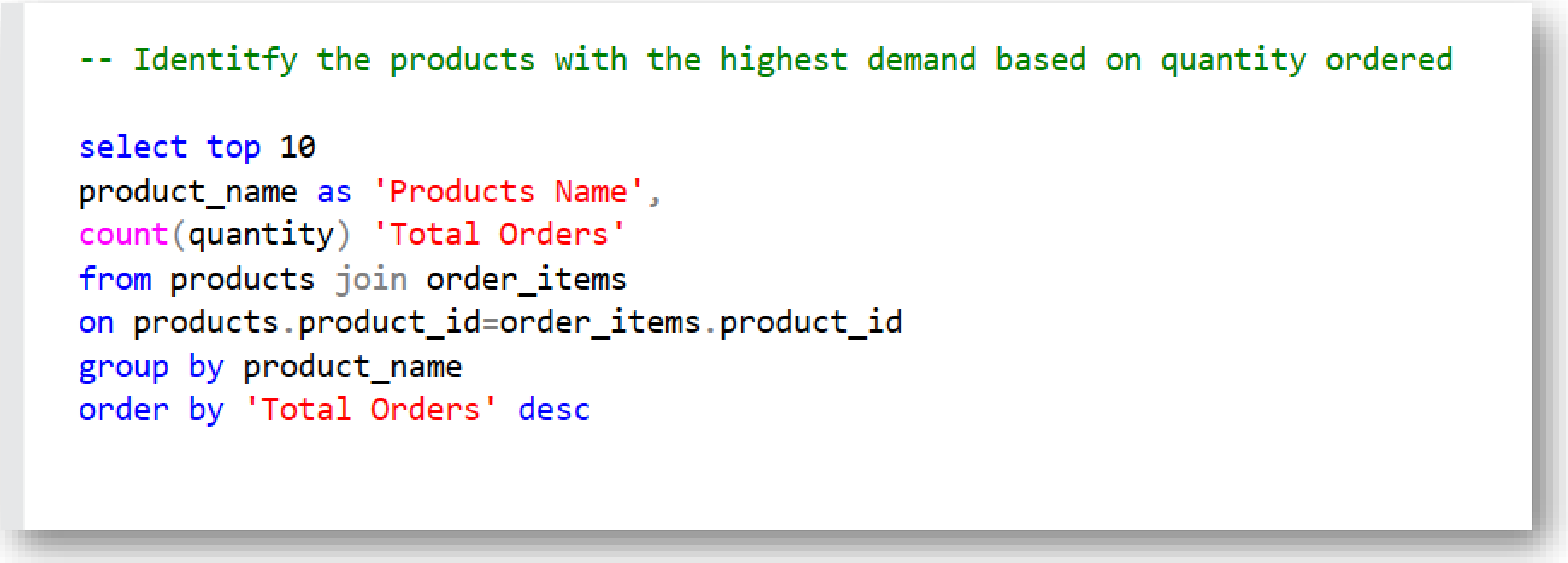
```
-- Calculate the total sales revenue for each products

select top 10
product_name as 'Products Name',
round(sum(total_price),2) as 'Total Revenue'
from products join order_items
on products.product_id=order_items.product_id
group by product_name
order by 'Total Revenue' desc
```

Output:



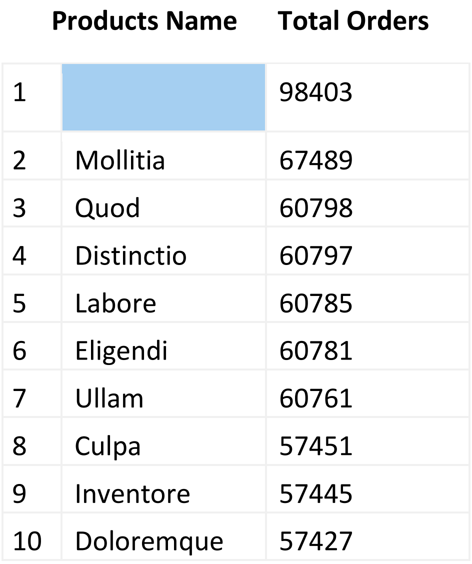
	Products Name	Total Revenue
1		49939599.88
2	Mollitia	34258044.55
3	Distinctio	30882253.17
4	Eligendi	30868670.88
5	Quod	30863060.22
6	Ullam	30859485.67
7	Labore	30857454.7
8	Culpa	29169371.15
9	Inventore	29161585.24
10	Illum	29159916.84

Query:



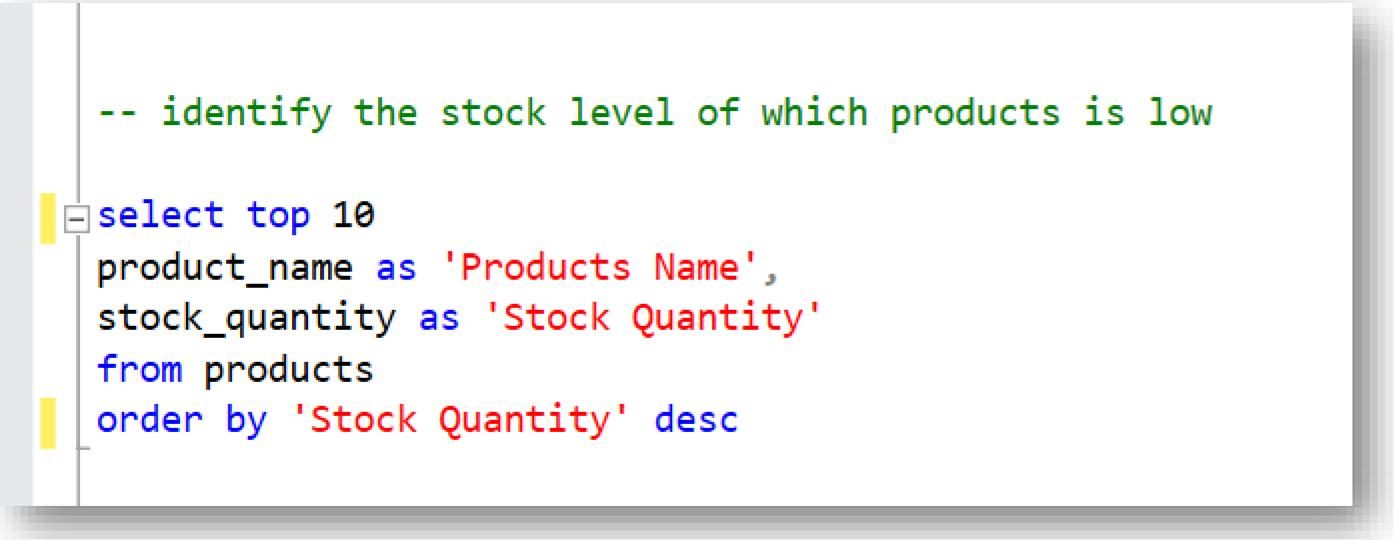
```sql
-- Identitfy the products with the highest demand based on quantity ordered

select top 10
product_name as 'Products Name',
count(quantity) 'Total Orders'
from products join order_items
on products.product_id=order_items.product_id
group by product_name
order by 'Total Orders' desc
```

Output:



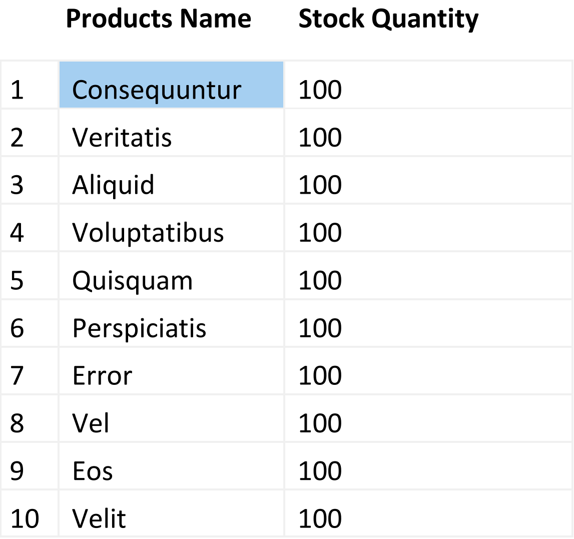
	Products Name	Total Orders
1		98403
2	Mollitia	67489
3	Quod	60798
4	Distinctio	60797
5	Labore	60785
6	Eligendi	60781
7	Ullam	60761
8	Culpa	57451
9	Inventore	57445
10	Doloremque	57427

Query:



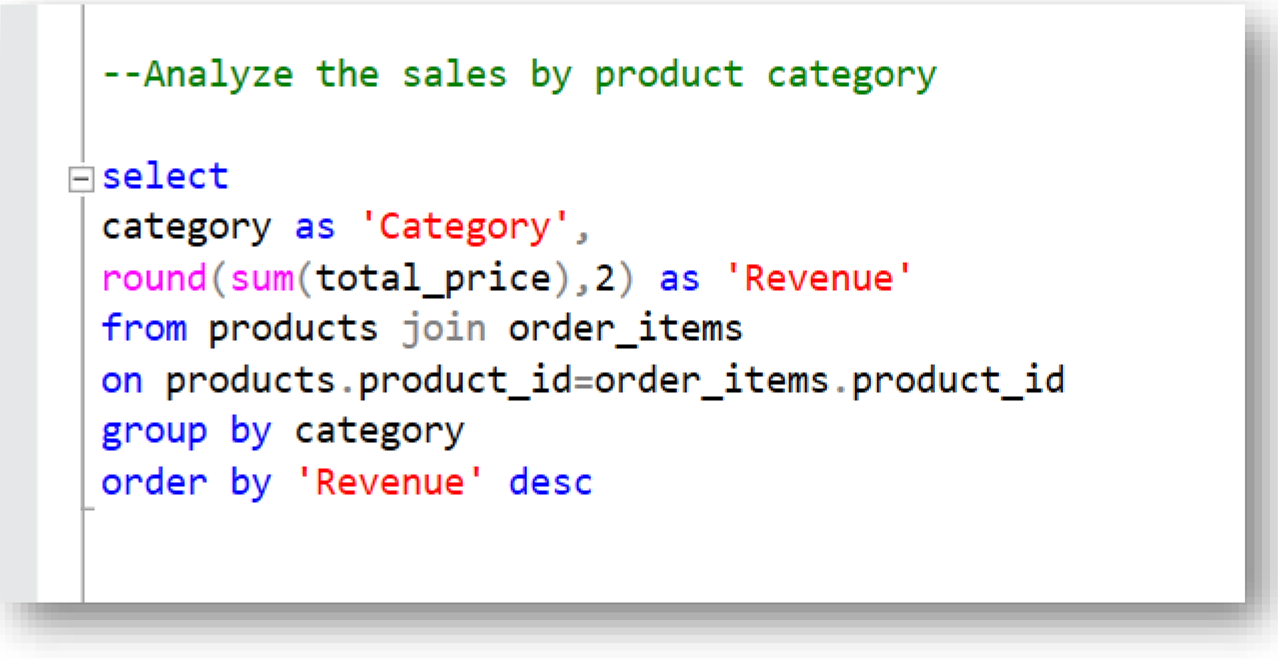
```sql
-- identify the stock level of which products is low

select top 10
product_name as 'Products Name',
stock_quantity as 'Stock Quantity'
from products
order by 'Stock Quantity' desc
```

Output:



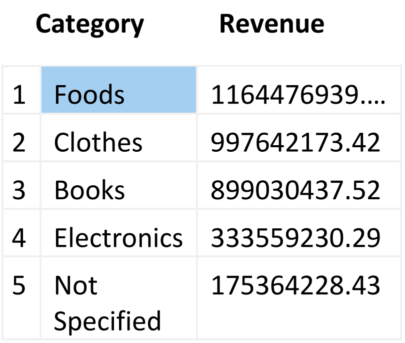
	Products Name	Stock Quantity
1	Consequuntur	100
2	Veritatis	100
3	Aliquid	100
4	Voluptatibus	100
5	Quisquam	100
6	Perspiciatis	100
7	Error	100
8	Vel	100
9	Eos	100
10	Velit	100

Query:



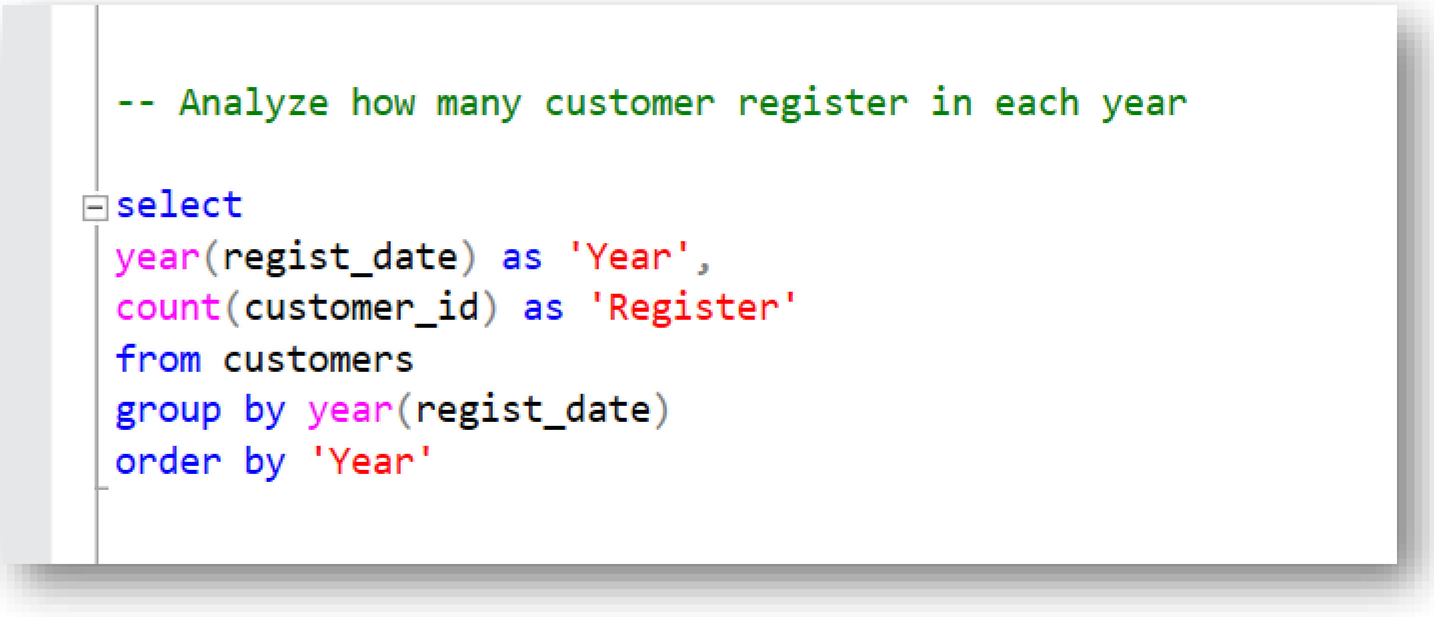
```sql
--Analyze the sales by product category

select
category as 'Category',
round(sum(total_price),2) as 'Revenue'
from products join order_items
on products.product_id=order_items.product_id
group by category
order by 'Revenue' desc
```

Output:



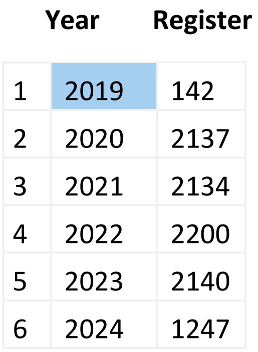
	Category	Revenue
1	Foods	1164476939....
2	Clothes	997642173.42
3	Books	899030437.52
4	Electronics	333559230.29
5	Not Specified	175364228.43

Query:



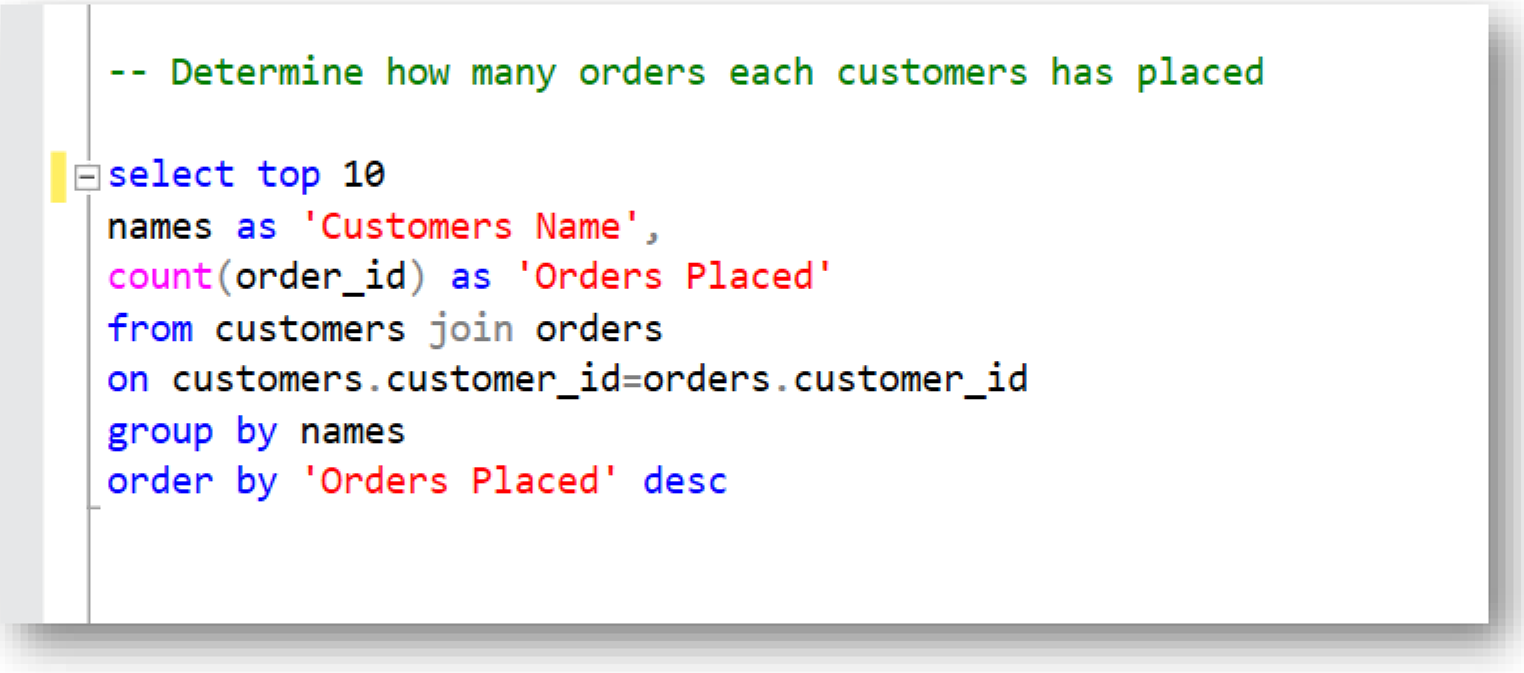
```
-- Analyze how many customer register in each year

select
year(regist_date) as 'Year',
count(customer_id) as 'Register'
from customers
group by year(regist_date)
order by 'Year'
```

Output:



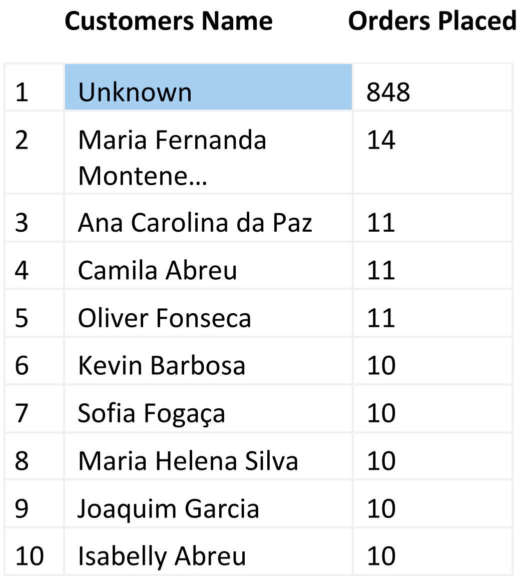
	Year	Register
1	2019	142
2	2020	2137
3	2021	2134
4	2022	2200
5	2023	2140
6	2024	1247

Query:



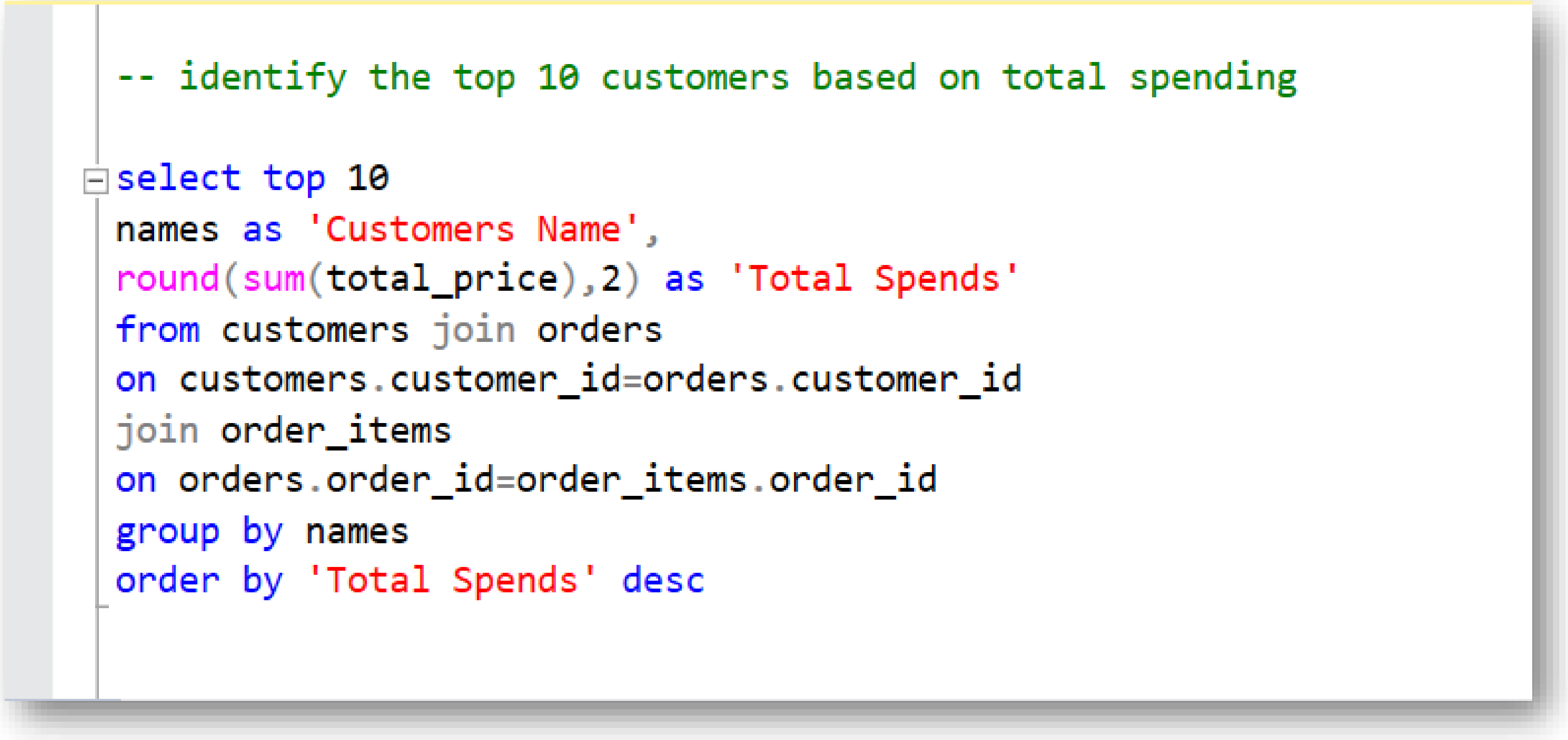
```
-- Determine how many orders each customers has placed

select top 10
names as 'Customers Name',
count(order_id) as 'Orders Placed'
from customers join orders
on customers.customer_id=orders.customer_id
group by names
order by 'Orders Placed' desc
```

Output:



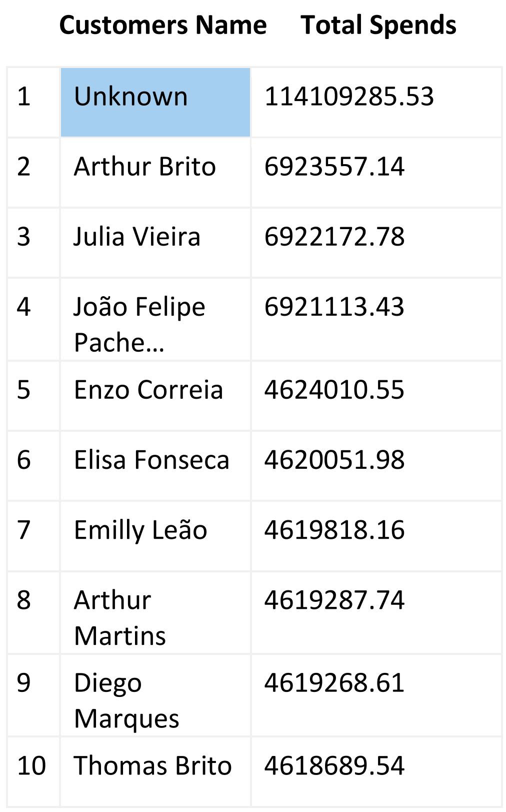
	Customers Name	Orders Placed
1	Unknown	848
2	Maria Fernanda Montene...	14
3	Ana Carolina da Paz	11
4	Camila Abreu	11
5	Oliver Fonseca	11
6	Kevin Barbosa	10
7	Sofia Fogaça	10
8	Maria Helena Silva	10
9	Joaquim Garcia	10
10	Isabelly Abreu	10

Query:



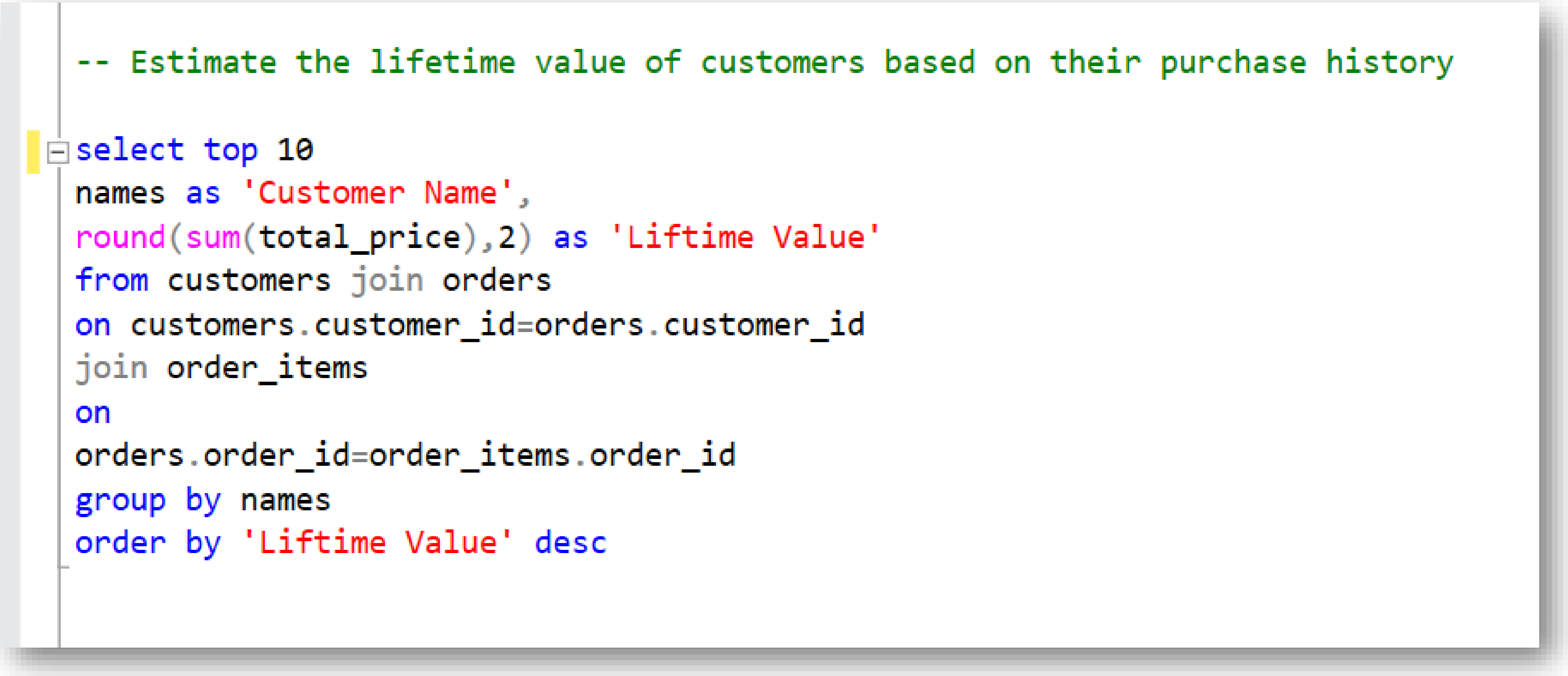
```
-- identify the top 10 customers based on total spending

select top 10
names as 'Customers Name',
round(sum(total_price),2) as 'Total Spends'
from customers join orders
on customers.customer_id=orders.customer_id
join order_items
on orders.order_id=order_items.order_id
group by names
order by 'Total Spends' desc
```

Output:



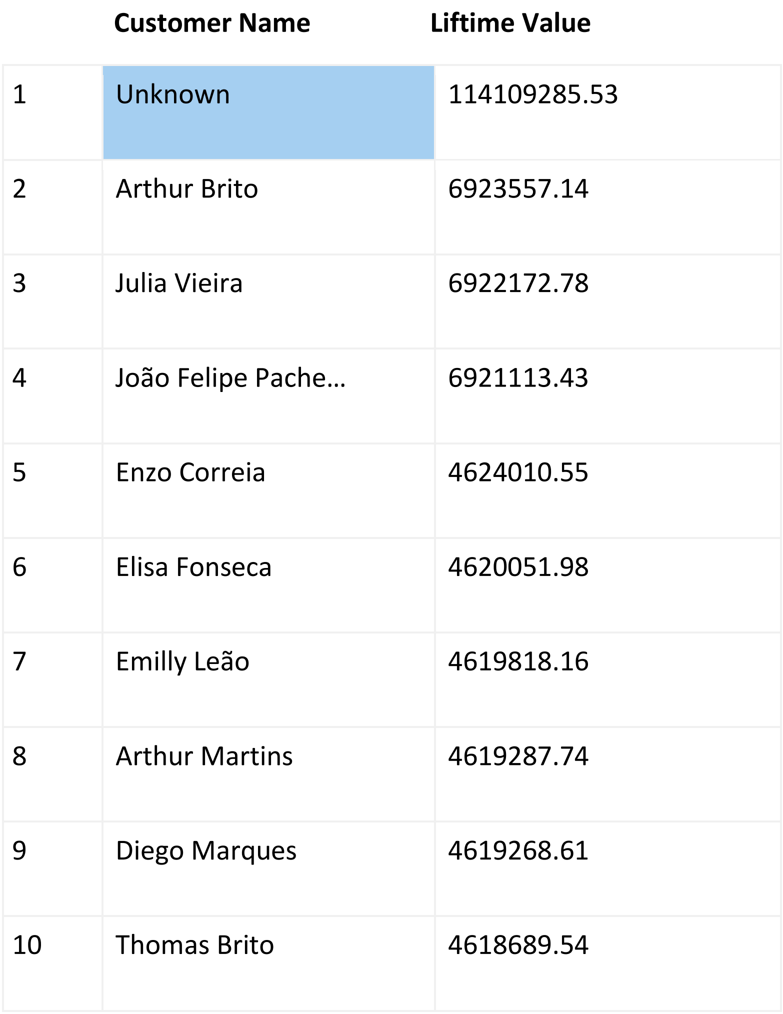
	Customers Name	Total Spends
1	Unknown	114109285.53
2	Arthur Brito	6923557.14
3	Julia Vieira	6922172.78
4	João Felipe Pache...	6921113.43
5	Enzo Correia	4624010.55
6	Elisa Fonseca	4620051.98
7	Emilly Leão	4619818.16
8	Arthur Martins	4619287.74
9	Diego Marques	4619268.61
10	Thomas Brito	4618689.54

Query:



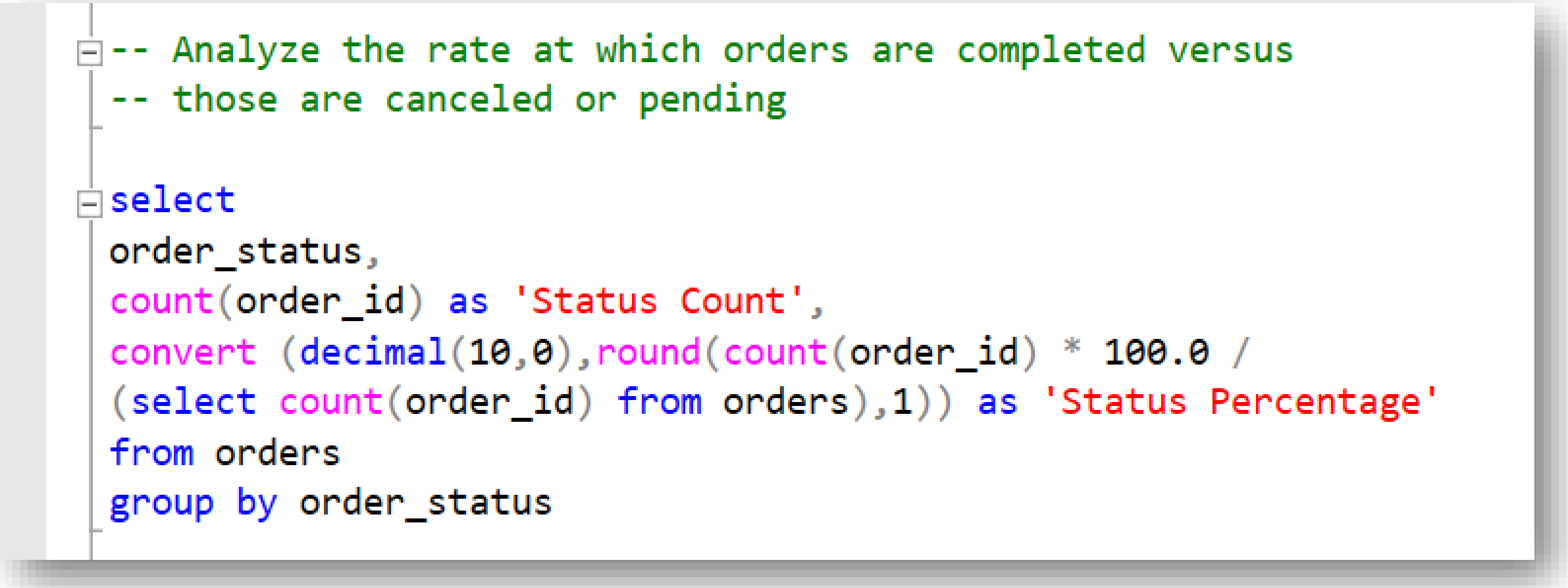
```sql
-- Estimate the lifetime value of customers based on their purchase history

select top 10
names as 'Customer Name',
round(sum(total_price),2) as 'Liftime Value'
from customers join orders
on customers.customer_id=orders.customer_id
join order_items
on
orders.order_id=order_items.order_id
group by names
order by 'Liftime Value' desc
```

Output:



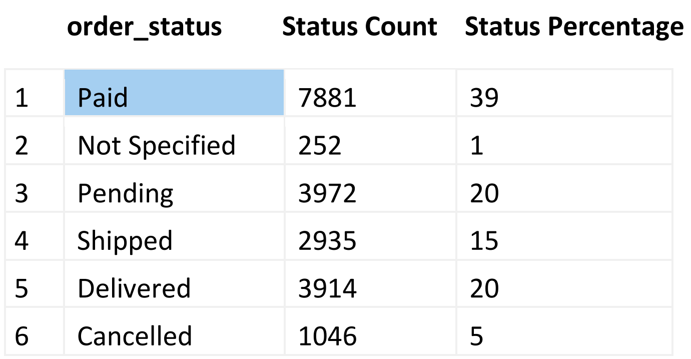
	Customer Name	Liftime Value
1	Unknown	114109285.53
2	Arthur Brito	6923557.14
3	Julia Vieira	6922172.78
4	João Felipe Pache...	6921113.43
5	Enzo Correia	4624010.55
6	Elisa Fonseca	4620051.98
7	Emilly Leão	4619818.16
8	Arthur Martins	4619287.74
9	Diego Marques	4619268.61
10	Thomas Brito	4618689.54

Query:



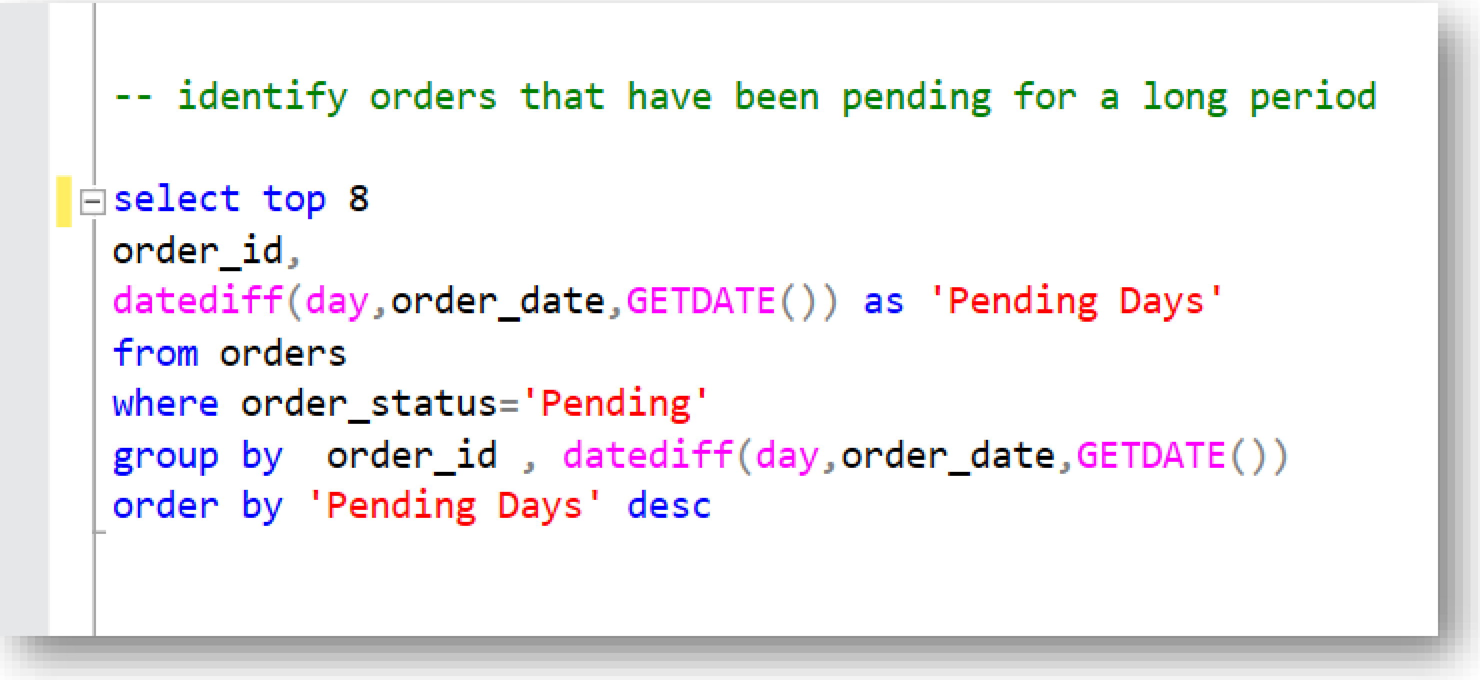
```sql
-- Analyze the rate at which orders are completed versus
-- those are canceled or pending

select
order_status,
count(order_id) as 'Status Count',
convert (decimal(10,0),round(count(order_id) * 100.0 /
(select count(order_id) from orders),1)) as 'Status Percentage'
from orders
group by order_status
```

Output:



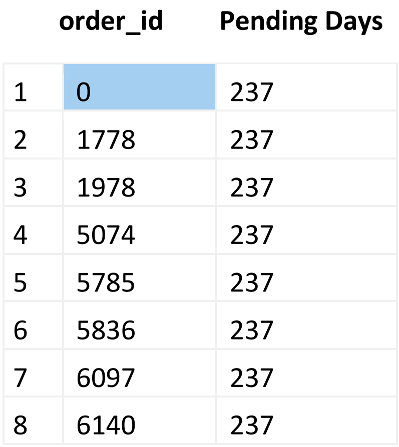
	order_status	Status Count	Status Percentage
1	Paid	7881	39
2	Not Specified	252	1
3	Pending	3972	20
4	Shipped	2935	15
5	Delivered	3914	20
6	Cancelled	1046	5

Query:



```
-- identify orders that have been pending for a long period

select top 8
order_id,
datediff(day,order_date,GETDATE()) as 'Pending Days'
from orders
where order_status='Pending'
group by  order_id , datediff(day,order_date,GETDATE())
order by 'Pending Days' desc
```

Output:



	order_id	Pending Days
1	0	237
2	1778	237
3	1978	237
4	5074	237
5	5785	237
6	5836	237
7	6097	237
8	6140	237