SQL Challenge

Introduction

In this case study, I have worked with a publicly available dataset, created the reports listed below, and answered the questions.

Create a query that provides the following information for all items ordered.

- OrderDetailID
- ProductName
- Product Quantity
- Unit Price
- Total Price for Product in Order
- CategoryName
- OrderID
- Order Date
- CustomerID
- Customer country
- Total Number of Items in Order
- Total Cost of the Order
- Employee Last Name that processed the order

```
select OrderDetailID, ProductName, Quantity, Price as  Unit_Price,
CategoryName,OrderID, OrderDate,sum(quantity) as total_items_in_order,
CustomerID, country, lastname, sum(Total_Price_for_Product) as
Total_Cost_of_Order, Total_Price_for_Product
from
(SELECT *, price*quantity as Total_Price_for_Product FROM [Orders] o
join orderdetails od
on od.OrderID= o.OrderID
join products p
on p.ProductID= od.ProductID
join categories ca
on ca.CategoryID= p.CategoryID
join customers c
on c.CustomerID= o.CustomerID
join employees e
on e.EmployeeID= o.EmployeeID) as f
group by OrderID
```

Result:

Number of Records: 196

OrderDetailID	ProductName	Quantity	Unit_Price	CategoryName	OrderID	OrderDate	total_items_in_order	CustomerID	Country	LastName	Total_Cost_of_Order	Total_Price_for_Product
1	Queso Cabrales	12	21	Dairy Products	10248	1996-07-04	27	90	Finland	Buchanan	566	252
4	Tofu	9	23.25	Produce	10249	1996-07-05	49	81	Brazil	Suyama	2329.25	209.25
6	Jack's New England Clam Chowder	10	9.65	Seafood	10250	1996-07-08	60	34	Brazil	Peacock	2267.25	96.5
9	Gustaf's Knäckebröd	6	21	Grains/Cereals	10251	1996-07-08	41	84	France	Leverling	839.5	126
12	Sir Rodney's Marmalade	40	81	Confections	10252	1996-07-09	105	76	Belgium	Peacock	4662.5	3240
15	Gorgonzola Telino	20	12.5	Dairy Products	10253	1996-07-10	102	34	Brazil	Leverling	1806	250
18	Guaraná Fantástica	15	4.5	Beverages	10254	1996-07-11	57	14	Switzerland	Buchanan	781.5	67.5
21	Chang	20	19	Beverages	10255	1996-07-12	110	68	Switzerland	Dodsworth	3115.75	380
25	Perth Pasties	15	32.8	Meat/Poultry	10256	1996-07-15	27	88	Brazil	Leverling	648	491.99999999999994
27	Schoggi Schokolade	25	43.9	Confections	10257	1996-07-16	46	35	Venezuela	Peacock	1400.5	1097.5
30	Chang	50	19	Beverages	10258	1996-07-17	121	20	Austria	Davolio	2529.75	950
33	Sir Rodney's Scones	10	10	Confections	10259	1996-07-18	11	13	Mexico	Peacock	126	100
35	Jack's New England Clam Chowder	16	9.65	Seafood	10260	1996-07-19	102	55	USA	Peacock	2183.9	154.4
39	Sir Rodney's Scones	20	10	Confections	10261	1996-07-19	40	61	Brazil	Peacock	560	200
41	Chef Anton's Gumbo Mix	12	21.35	Condiments	10262	1996-07-22	29	65	USA	Callahan	782.2	256.20000000000005
44	Pavlova	60	17.45	Confections	10263	1996-07-23	184	20	Austria	Dodsworth	3086.4	1047
48	Chang	35	19	Beverages	10264	1996-07-24	60	24	Sweden	Suyama	906.25	665
50	Alice Mutton	30	39	Meat/Poultry	10265	1996-07-25	50	7	France	Fuller	1470	1170
52	Queso Manchego La Pastora	12	38	Dairy Products	10266	1996-07-26	12	87	Finland	Leverling	456	456

Question 2

Please create the following smaller reports:

- 🎬 Which employee has sold the most products?

SQL Statement:

```
select count(ORDERID) as highest_order,EmployeeID  from orders
GROUP BY EmployeeID
order by highest_order desc
limit 1
;
```

Edit the SQL Statement, and click "Run SQL" to see the result.

Run SQL »

Result:

Number of Records: 1

highest_order	EmployeeID
40	4

🎬 Which product has generated the most revenue for the business?

SQL Statement:

```
select p.productname, sum(p.price*o.quantity)as highest_revenue from orderdetails o
join products p
on p.productid= o.productid
group by productname
order by highest_revenue desc
limit 1;
```

Edit the SQL Statement, and click "Run SQL" to see the result.

Run SQL »

Result:

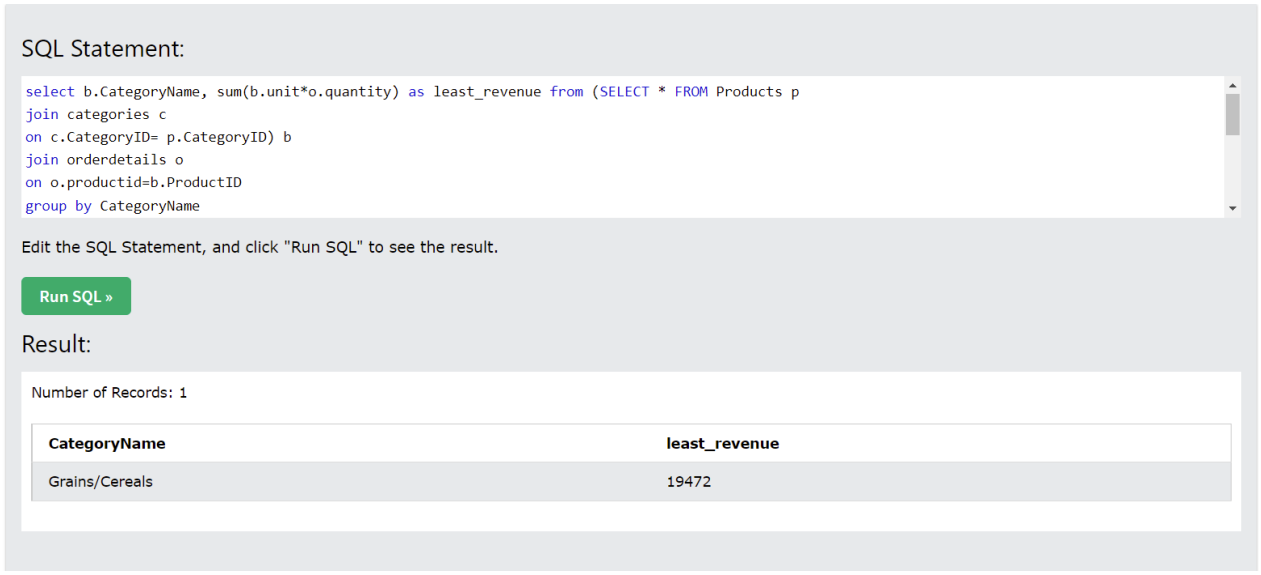
Number of Records: 1

ProductName	highest_revenue
Côte de Blaye	62976.5

select p.productname, sum(p.price*o.quantity)as highest_revenue from orderdetails o
join products p
on p.productid= o.productid
group by productname
order by highest_revenue desc
limit 1;

🎬 **Which category has produced the least revenue for the business?**

```
select b.CategoryName, sum(b.unit*o.quantity) as least_revenue from (SELECT *
FROM Products p
join categories c
on c.CategoryID= p.CategoryID) b
join orderdetails o
on o.productid=b.ProductID
group by CategoryName
order by least_revenue asc
limit 1;
```



🎬 **How many customers have ordered once, more than once, or never?**

```
select sum(case
when number_of_orders_placed= 0 then number_of_customers
End) as Never_ordered,
sum(case when number_of_orders_placed= 1 then
number_of_customers End) as Ordered_once,
sum(case when number_of_orders_placed>1 then
number_of_customers End) as more_than_once
from(
select count(num_orderid) as number_of_customers, num_orderid
as number_of_orders_placed
from
(select c.CustomerID, o.CustomerID, count(distinct orderid) as
num_orderid
 from customers c
 left join orders o
 on o.CustomerID= c.CustomerID
 group by c.CustomerID) as a
 group by num_orderid);
```

🎬 Which employee has shipped the most products?

SQL Statement:

```
SELECT count(od.Quantity) as most_products, o.EmployeeID FROM [Orders] as o
join OrderDetails od
on od.OrderID=o.OrderID
group by EmployeeID
order by most_products desc
limit 1;
```

Edit the SQL Statement, and click "Run SQL" to see the result.

Run SQL »

Result:

Number of Records: 1

most_products	EmployeeID
123	4

SELECT count(od.Quantity) as most_products, o.EmployeeID
FROM [Orders] as o
join OrderDetails od
on od.OrderID=o.OrderID
group by EmployeeID
order by most_products desc
limit 1;

- **What percentage of orders by $ are Dairy products as a percentage of all order value by employees?**

```
select employeeid, ((total_dairy/total)*100) as percentage
from
(select sum(case when CategoryName= 'Dairy Products' then (price*Quantity) End)
    as total_dairy, sum(price*Quantity) as total, employeeid
from orderdetails od
join orders o
on o.orderid=od.orderid
join Products p
on p.ProductID=od.ProductID
join categories c
on c.CategoryID= p.CategoryID
group by employeeid)
```

- Which employee has shipped the most orders over $1500 by count to a country that has a vowel as its fourth letter?

```
SELECT *, sum( price* Quantity) FROM [OrderDetails] od
join orders o
on o.orderid=od.orderid
join products p
on od.ProductID= p.ProductID
group by o.orderid
group by EmployeeID

(SELECT *, count(o.orderid), sum( price* Quantity) FROM [OrderDetails] od
join orders o
on o.orderid=od.orderid
join products p
on od.ProductID= p.ProductID
group by od.orderid)



SELECT *, count(o.orderid), sum( price* Quantity) FROM [OrderDetails] od
join orders o
on o.orderid=od.orderid
join products p
on od.ProductID= p.ProductID
join customers c
on c.CustomerID=o.CustomerID
where Country='Finland'
group by od.orderid
```

```
SELECT *, count(o.orderid), sum( price* Quantity) FROM [OrderDetails] od
join orders o
on o.orderid=od.orderid
join products p
on od.ProductID= p.ProductID
join customers c
on c.CustomerID=o.CustomerID
where lower(substr(country,4,1)) in ('a','e','i','o','u')
group by od.orderid
```

```
select *, count(employee_order) as most_order from
(SELECT *, count(o.orderid) as employee_order, sum( price* Quantity) FROM
    [OrderDetails] od
join orders o
on o.orderid=od.orderid
join products p
on od.ProductID= p.ProductID
join customers c
on c.CustomerID=o.CustomerID
where lower(substr(country,4,1)) in ('a','e','i','o','u')
group by od.orderid) as b
group by employeeid
```

```
select b.EmployeeID, count(b.employee_order) as total_orders,
    sum(b.product_price_sold) as total_order_cost, country
from
(SELECT *, count(o.orderid) as employee_order, sum(price* Quantity) as
    product_price_sold FROM [OrderDetails] od
join orders o
on o.orderid=od.orderid
join products p
on od.ProductID= p.ProductID
join customers c
on c.CustomerID=o.CustomerID
where lower(substr(country,4,1)) in ('a','e','i','o','u')
group by od.orderid ) as b
```

```
group by b.EmployeeID
```

- For customers with more than one order, what was the amount of time elapsed between their last order and the one prior to that?

```
with multiple_order as(
select CustomerID, count(distinct OrderID) as no_of_orders
from orders
group by CustomerID
having no_of_orders> 1
order by CustomerID asc
)
select o.orderdate, o.customerid, o.orderid
from multiple_order mo
inner join orders o
on mo.CustomerID=o.CustomerID
```

```
with multiple_order as(
select CustomerID, count(distinct OrderID) as no_of_orders
from orders
group by CustomerID
having no_of_orders> 1
order by CustomerID asc
)
select o.orderdate, o.customerid, o.orderid, row_number() over (
partition by o.CustomerID order by o.orderdate asc) as rn
from
multiple_order mo
inner join orders o
on mo.CustomerID=o.CustomerID
```

```
SELECT CustomerID,  count(OrderID) FROM [Orders]
group by CustomerID
having count(OrderID) > 1
```

```
select OrderDate, CustomerID,OrderID,count(CustomerID) from orders
group by CustomerID
having count(OrderID)> 1
```

```
order by CustomerID asc
```

- Bonus question:
  - How many orders has the Batman sold to customers whose names end in a consonant?

```
SELECT em.EmployeeID, em.LastName, em.FirstName,
    COUNT(Customers.CustomerID) as  'number of orders'
FROM (SELECT *
FROM Employees
LEFT JOIN Orders ON Employees.EmployeeID = Orders.EmployeeID
WHERE Employees.EmployeeID = 10) as em
LEFT JOIN Customers ON em.CustomerID = Customers.CustomerID
WHERE Customers.CustomerID is null
OR (Customers.CustomerName not LIKE '%a'
   AND Customers.CustomerName not LIKE '%e'
   AND Customers.CustomerName not LIKE '%i'
   AND Customers.CustomerName not LIKE '%o'
   AND Customers.CustomerName not LIKE '%u'
   AND Customers.CustomerName not LIKE '%y')
```

```
Select b.EmployeeID, b.LastName, b.FirstName,
COUNT(c.CustomerID) as  no_of_orders
From (SELECT *
From Employees e
LEFT JOIN Orders o
ON e.EmployeeID = o.EmployeeID
WHERE e.EmployeeID = 10) as b
LEFT JOIN Customers c
ON b.CustomerID = c.CustomerID
WHERE c.CustomerID is null
or (c.CustomerName not LIKE '%a'
   and c.CustomerName not LIKE '%e'
   and c.CustomerName not LIKE '%i'
   and c.CustomerName not LIKE '%o'
   and c.CustomerName not LIKE '%u'
   and c.CustomerName not LIKE '%y')
```

```
SELECT Employees.FirstName, Employees.LastName,
    COUNT(Orders.EmployeeID) as "Number of Sales"
FROM Employees
LEFT JOIN Orders ON Employees.EmployeeID = Orders.EmployeeID
GROUP BY Employees.EmployeeID
```

Question 3

Please generate a report that shows how many items from each category each employee has sold, showing the employee name, product name, and quantity sold

```
select sum(Quantity),CategoryID, EmployeeID from

(SELECT * FROM [OrderDetails] od

join products p

on p.ProductID=od.ProductID

join orders o

on o.OrderID= od.OrderID) as b

group by CategoryID
```

```
selECT *,  sum(quantity) FROM [OrderDetails] od

join products p

on p.ProductID=od.ProductID

join orders o

on o.OrderID= od.OrderID

group by CategoryID, EmployeeID
```

```
select sum(Quantity), CategoryID, EmployeeID
from
(SELECT * FROM [OrderDetails] od
join products p
on p.ProductID=od.ProductID
join orders o
on o.OrderID= od.OrderID
group by CategoryID, EmployeeID) as b
group by CategoryID, EmployeeID
```