- **The question number: 1**
  Show the top 5 countries with the highest number of Customers. However, do not include customers from Oregon (OR) or Texas (TX) in the counts for USA.
- **The SQL query that you created to answer the question**

```
select
        c.country,
        count(distinct c.customerid) as numberofcustomers
from customers c
left join
        (
        select c2.customerid
        from customers c2
        where
                c2.country = 'USA'
                and c2.region in ('OR', 'TX')
        ) as subquery
on c.customerid = subquery.customerid
where subquery.customerid is null
group by c.country
order by numberofcustomers desc
limit 5;
```

- **A screenshot of the output that your query produces.**

	A-Z country	123 numberofcustomers
1	Germany	11
2	France	11
3	Brazil	9
4	USA	9
5	UK	7

- **The question number: 2**
  Find the total number of products in each category of products. Show the categories with the most products first.
- **The SQL query that you created to answer the question**

```
select
        c.categoryname,
        count(p.productid) as totalproducts
from products p
```

```
join categories c
on p.categoryid = c.categoryid
group by c.categoryname
order by totalproducts desc;
```

- **A screenshot of the output that your query produces.**

	A-Z categoryname	123 totalproducts
1	Confections	13
2	Condiments	12
3	Beverages	12
4	Seafood	12
5	Dairy Products	10
6	Grains/Cereals	7
7	Meat/Poultry	6
8	Produce	5

- **The question number: 3**
  Northwind Traders will prioritize re-ordering a certain product if: (1) UnitsInStock plus UnitsOnOrder are less than or equal to ReorderLevel, (2) the product is not discontinued (Discontinued = 0), and (3) the product has been ordered more than 1 time. Which products need to be prioritized for reordering? Only display product id and name and sort output by product id.
- **The SQL query that you created to answer the question**

```
select
        p.productid,
        p.productname
from products p
join orderdetails as od
on p.productid = od.productid
where
        (p.unitsinstock + p.unitsonorder) <= p.reorderlevel
        and p.discontinued = 0
group by p.productid
having count(od.orderid) > 0;
```

- **A screenshot of the output that your query produces.**

	productid	productname
1	30	Nord-Ost Matjeshering
2	70	Outback Lager

- **The question number: 4**
  How many orders were shipped by United Package and Speedy Express (combined) in 1997?
- **The SQL query that you created to answer the question**

```
select count(o.orderid) as order1997 from orders o
join shippers s
on o.shipvia = s.shipperid
where
        s.companyname in ('United Package', 'Speedy Express')
        and o.shippeddate between '1997-01-01' and '1997-12-31';
```

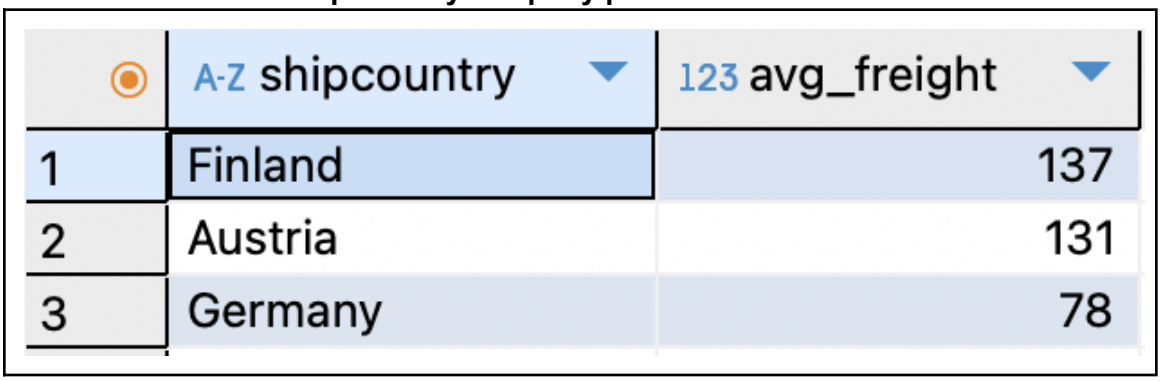
- **A screenshot of the output that your query produces.**

	order1997
1	273

- **The question number: 5**
  Northwind's operations team has noted the high cost of freight charges for Speedy Express shipments. For 1996, return the three ship countries with the highest average freight, in descending average freight order, for orders fulfilled by Speedy Express.
- **The SQL query that you created to answer the question**

```
select
        o.shipcountry,
        round(avg(o.freight)) as avg_freight
from orders o
join shippers s
on o.shipvia = s.shipperid
where
        s.companyname = 'Speedy Express'
        and o.shippeddate between '1996-01-01' and '1996-12-31'
group by o.shipcountry
order by avg_freight desc
limit 3;
```

- **A screenshot of the output that your query produces.**

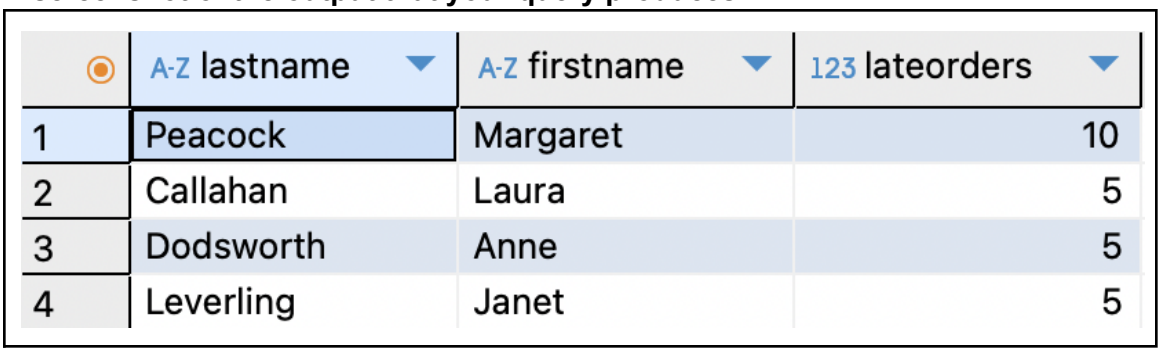


	A-Z shipcountry	123 avg_freight
1	Finland	137
2	Austria	131
3	Germany	78

- **The question number: 6**
  Some salespeople have more orders arriving late than other salespeople ("late" defined as RequiredDate <= ShippedDate). Which salespeople have at least 5 orders arriving late? Do not show any employees with less than 5 late orders and sort output starting with the employee that has the most late orders.
- **The SQL query that you created to answer the question**

```
select
        e.lastname,
        e.firstname,
        count(o.orderid) as lateorders
from orders o
join employees e
on o.employeeid = e.employeeid
where o.requireddate <= o.shippeddate
group by e.lastname, e.firstname
having count(o.orderid) >= 5
order by lateorders desc;
```

- **A screenshot of the output that your query produces.**



	A-Z lastname	A-Z firstname	123 lateorders
1	Peacock	Margaret	10
2	Callahan	Laura	5
3	Dodsworth	Anne	5
4	Leverling	Janet	5

- **The question number: 7**
  Suppose that Northwind wants to send all of the high-value customers a special VIP gift. A high-value customer is anyone who've made at least 1 order with a total value (quantity x unit price) equal to $10,000 or more. Find all of these high-value customers for 1996.

- **The SQL query that you created to answer the question**

```
select
        c.companyname,
        o.orderid,
        round(sum(od.quantity * od.unitprice)) as totalorderamount
from orders o
join orderdetails od
on o.orderid = od.orderid
join customers c
on o.customerid = c.customerid
where o.orderdate between '1996-01-01' and '1996-12-31'
group by
        c.companyname,
        o.orderid
having round(sum(od.quantity * od.unitprice)) >= 10000
order by totalorderamount desc;
```

- **A screenshot of the output that your query produces.**

	companyname	orderid	totalorderamount
1	Queen Cozinha	10,372	12,281
2	Piccolo und mehr	10,353	10,742

- **The question number: 8**
  Which customers never placed any order?
- **The SQL query that you created to answer the question**

```
select c.companyname
from customers c
left join orders o
on c.customerid = o.customerid
where o.orderid is null;
```

- **A screenshot of the output that your query produces.**

	companyname
1	Paris spécialités
2	FISSA Fabrica Inter. Salchichas S.A.

- **The question number: 9**

Which employees and which territories are not assigned yet (to territories and employees respectively), if any.

- **The SQL query that you created to answer the question**

```
select
        e.employeeid,
        t.territoryid
from employees e
full join employeeterritories et
on e.employeeid = et.employeeid
full join territories t
on t.territoryid = et.territoryid
where
        e.employeeid is null
        or t.territoryid is null;
```

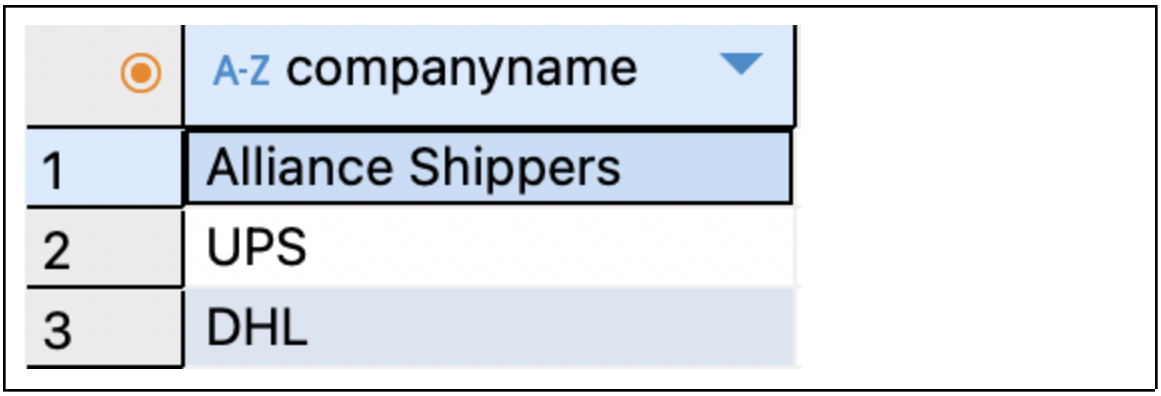
- **A screenshot of the output that your query produces.**

	employeeid	territoryid
1	[NULL]	75234
2	[NULL]	72716
3	[NULL]	78759
4	[NULL]	29202

- **The question number: 10**
  What are the shippers that Northwind did not work with?
- **The SQL query that you created to answer the question**

```
select s.companyname
from shippers s
left join orders o
on s.shipperid = o.shipvia
where o.orderid is null
order by s.shipperid;
```

- **A screenshot of the output that your query produces.**

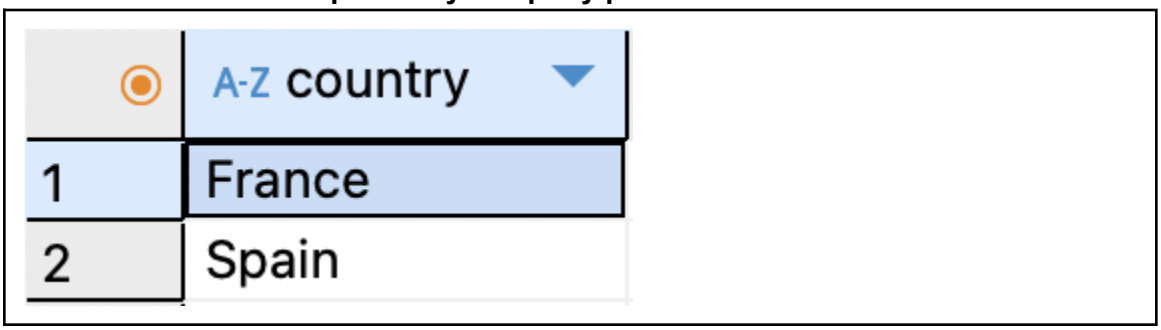


	A-Z companyname
1	Alliance Shippers
2	UPS
3	DHL

- **The question number: 11**
  Produce a report showing a list of countries where there are customers who have never placed an order. In your answer, you should not use FULL JOIN. Also, make sure that your query (and your output) would be correct for any data, not only the specific data that we happen to have here.
- **The SQL query that you created to answer the question**

```
select distinct c.country from orders o
right join customers c
on o.customerid = c.customerid
where o.orderid is null;
```

- **A screenshot of the output that your query produces.**



	A-Z country
1	France
2	Spain

- **The question number: 12**
  How many orders were processed by every Northwind employee? In your answer, you should not use FULL JOIN. Also, make sure that your query (and your output) would be correct for any data, not only the specific data that we happen to have here.
- **The SQL query that you created to answer the question**

```
select
        e.employeeid,
        count(o.orderid) as orders_per_emp
from employees e
left join orders o
on e.employeeid = o.employeeid
group by e.employeeid
```

```
order by e.employeeid;
```

- **A screenshot of the output that your query produces.**

	employeeid	orders_per_emp
1	1	123
2	2	96
3	3	127
4	4	156
5	5	42
6	6	67
7	7	72
8	8	104
9	9	43