1. There are several situations that can cause the average wrong: error of the input, missing of data, etc. So, in this case, AOV is much higher than expected. Maybe it is some shops submit wrong order amount numbers, or wrong prices. Since AOV is always the total revenue divided by number of orders, I calculate AOV per order, and sort the whole list with AOV from high to low. It not hard to find the shop 78 looks weird. It sells each item for $25725, while the next is $352 by shop 42. $25725 is too high for sneaker, it must be wrong. When I hide the data from shop 78, and calculate the new AOV, it is around $307.01, much more reasonable.

So, I think the shop 78 should be checked for correctness of data, and the new AOV is around $307.01 before the shop 78’s result comes out. I get this result by using AOV per shop as the metric.

(By the way, I think this data is too fake. Maybe I am wrong here. But I find the same user id appears in different shops many times in a single month and accept huge difference in price. I think if this really happens, all shops should learn something from shop 78. Who don’t want their AOV to be higher when consumers will accept price variance?)

1. (a) I check the shipper list and find Speedy Express’s shipper id is ‘1’. Then go to the order list and try to find out how many shipper id is ‘1’. And it turns out to be 54. So there are 54 orders shipped by Speedy Express in total.

Run code:

SELECT count(ShipperID)

FROM [Orders]

WHERE ShipperID in (

SELECT ShipperID

FROM [Shippers]

WHERE ShipperName='Speedy Express');

(b)I find the most frequent employee id in order list and find the last name related to this id in employee list.

Run code:

Select lastname

From [Employees]

Where EmployeeID in (

Select EmployeeID

From [Orders]

Group BY EmployeeID

Order BY COUNT(\*) DESC

Limit 1);

(c)I first find the consumer IDs which represents for the consumers in Germany in consumer list, then I find the order with these consumer IDs in order list and find the most frequent products id in order-detail list. Return the product name by product id in product list.

Run code:

Select distinct ProductID From [OrderDetails] Where OrderID in (

Select OrderID FROM [Orders] WHERE CustomerID in (

Select CustomerID FROM [Customers] WHERE Country='Germany'))

Group BY ProductID

ORDER BY sum(QUANTITY) DESC

LIMIT 1;