### 1A

From observing the dataset and exploratory data analysis, each line refers to one order placed by a customer, which may include multiple pairs of shoes.

Additionally, there seems to be an outlier for shop 78 as their 1 shoe order was priced at 25725 which would have skewed our calculation, assuming it is faulty data since 25725 is an unlikely price for a pair of shoes.

It would appear that the AOV calculation currently is done as an aggregate for all sneaker shops taking total revenue divided by total number of orders (5000), which does not communicate much in terms of individual shops’ performance and would not be useful for actionable steps for each respective shop.

However, this information could definitely be useful for Shopify’s internal analysis on overall consumer spending in the category.

A better way to evaluate the data and give more segregated insights would be to calculate AOV per shop using the same Total Order Amount/ Orders but for each shop.

### 1B

Aside from the AOV for each shop, a metric I would report from the dataset is the **average shoe price across each shop**. This could be used by individual shops as a measure to position their brand and perceived value to customers.

### 1C

Assuming order amounts do not take into account promotions and discounts,

If we take into account the outlier (Shop 78) then the average shoe price would be: **$407.99**

If we don’t take into account the outlier, then the average shoe price would be: **$152.26**

### 2A

SELECT COUNT (Orders.OrderID) AS 'Total Orders by Speedy Express'

FROM Shippers, Orders

WHERE Shippers.ShipperID = Orders.ShipperID AND

Shippers.ShipperID = 1

**SOLUTION: 54**

### 2B

SELECT TOP 1 \*

FROM (

SELECT Employees.LastName, Count(\*)

FROM Employees, Orders

WHERE Employees.EmployeeID = Orders.EmployeeID

GROUP BY Employees.LastName

Order by Count(\*) Desc)

**SOLUTION: Peacock (40 orders)**

### 2C

SELECT TOP 1 \*

FROM (

SELECT Products.ProductName, Count(\*)

FROM Orders, OrderDetails, Customers, Products

WHERE Orders.OrderID = OrderDetails.OrderID AND

Customers.Country = 'Germany' AND

Products.ProductID = OrderDetails.ProductID AND

Orders.CustomerID = Customers.CustomerID

GROUP BY Products.ProductName

Order by Count(\*) Desc)

SOLUTION: Gorgonzola Telino (ordered 5 times)