

Fall 2022 Data Science Intern Challenge

Please complete the following questions, and provide your thought process/work. You can attach your work in a text file, link, etc. on the application page. Please ensure answers are easily visible for reviewers!

Question 1: Given some sample data, write a program to answer the following: [click here to access the required data set](#)

On Shopify, we have exactly 100 sneaker shops, and each of these shops sells only one model of shoe. We want to do some analysis of the average order value (AOV). When we look at orders data over a 30 day window, we naively calculate an AOV of \$3145.13. Given that we know these shops are selling sneakers, a relatively affordable item, something seems wrong with our analysis.

- a. Think about what could be going wrong with our calculation. Think about a better way to evaluate this data.

The issue with this data which makes the AOV a poor metric for understanding the size of most orders is that there are a few extremely large orders (in terms of dollar amount) which makes the AOV an order of magnitude larger than the median.

Also, the store with store_id 78 appears to be selling shoes for over \$25,000 per pair, which is very high and may warrant further investigation.

- b. What metric would you report for this dataset?

The median would be a better representative value for order size.

- c. What is its value?

The median value is \$284.

Question 2: For this question you'll need to use SQL. [Follow this link](#) to access the data set required for the challenge. Please use queries to answer the following questions. Paste your queries along with your final numerical answers below.

- a. How many orders were shipped by Speedy Express in total?

Speedy Express shipped 54 orders.

```
SELECT COUNT(OrderID) as NumberOfOrders
FROM Orders WHERE ShipperID IS (
    SELECT ShipperID
    FROM Shippers
    WHERE ShipperName IS "Speedy Express");
```

- b. What is the last name of the employee with the most orders?

The last name of the employee with the most orders is Peacock.

Note that the code below accounts for the possibility of more than one employee having the maximum number of orders and returns all employees with maximum orders.

```
WITH EmployeeOrders AS (
    SELECT EmployeeID, COUNT(Orders.OrderID) as NumberOfOrders
    FROM Orders
    GROUP BY EmployeeID
),
MaxOrders AS (
    SELECT MAX (NumberOfOrders) as NumberOfOrders
    FROM EmployeeOrders
),
EmployeesWithMaxOrders AS (
    SELECT EmployeeID
    FROM EmployeeOrders
    INNER JOIN MaxOrders
    ON EmployeeOrders.NumberOfOrders = MaxOrders.NumberOfOrders
)
SELECT LastName
FROM Employees
INNER JOIN EmployeesWithMaxOrders
ON Employees.EmployeeID = EmployeesWithMaxOrders.EmployeeID;
```

- c. What product was ordered the most by customers in Germany?

The product that was ordered the most by customers in Germany is Boston Crab Meat.

This answer assumes that the most ordered product is based on quantity ordered, rather than number of orders placed. Note that this code accounts for the possibility of more than one product being ordered the maximum number of times and will return all such products.

```
WITH GermanCustomerID AS (
    SELECT CustomerID
    FROM Customers
    WHERE Country = "Germany"
),
GermanOrderID AS (
    SELECT Orders.OrderID
    FROM Orders
    INNER JOIN GermanCustomerID
    ON Orders.CustomerID = GermanCustomerID.CustomerID
),
ProductsOrderedInGermany AS (
    SELECT OrderDetails.ProductID, SUM(OrderDetails.Quantity) as TotalQuantity
    FROM GermanOrderID
    INNER JOIN OrderDetails
    ON GermanOrderID.OrderID = OrderDetails.OrderID
    GROUP BY OrderDetails.ProductID
),
MaxQuantity AS (
    SELECT MAX (TotalQuantity) as TotalQuantity
    FROM ProductsOrderedInGermany
),
MostOrderedProductID AS (
    SELECT ProductID
    FROM ProductsOrderedInGermany
    INNER JOIN MaxQuantity
    ON ProductsOrderedInGermany.TotalQuantity = MaxQuantity.TotalQuantity
)
SELECT Products.ProductName
FROM Products
INNER JOIN MostOrderedProductID
ON Products.ProductID = MostOrderedProductID.ProductID;
```