Solutions to Shopify's Fall 2022 Data Science Intern Challenge

By

Damilola-Oluwa Adeniyi

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.
> b. What metric would you report for this dataset?
> c. What is its value?

Solution to Question 1:

The solution to Question 1 was completed in a Jupyter Notebook that is hosted on a Colaboratory which can be found at this link: Solution to Question 1

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?
b. What is the last name of the employee with the most orders?
c. What product was ordered the most by customers in Germany?

Solution to Question 2:

a. SELECT COUNT(OrderID) AS OrdersShippedBySpeedyExpress
FROM Orders AS O
LEFT JOIN Shippers AS S
ON O.ShipperID == S.ShipperID
WHERE S.ShipperName = 'Speedy Express';

FINAL ANSWER: 54

This solution takes into consideration the joining of two tables – Orders and Shippers. A left join is made use of. A left join returns a table of all rows in the left table and the matching rows from the right table. In this case, after the joining of both tables on the appropriate column (ShipperID), a table where the ShipperName is Speedy Express will be returned as this was stated in the WHERE clause of the above query. A count is done on the OrderID to give a result of the total orders that were shipped by Speedy Express.

b. SELECT E.LastName AS LastNameOfEmployeeWithMostOrders,
COUNT(O.OrderID) AS MostOrders
FROM Employees AS E
LEFT JOIN Orders AS O
ON E.EmployeeID == O.EmployeeID
GROUP BY E.LastName
ORDER BY MostOrders DESC LIMIT 1;

FINAL ANSWER: Peacock

This solution takes into consideration the joining of two tables – Employees and Orders. A left join is made use of. A left join returns a table of all rows in the left table and the matching rows from the right table. In this case, after the joining of both tables on the appropriate column (EmployeeID), a table with Last Names and Order IDs was returned. This table has duplicating last names as more than one order has been made by an employee. To obtain the last name of the employee with the most orders, a count was done on the Order IDs and the table

was grouped by the employee's last name and ordered in a descending order to get the highest count. Futhermore, a limit of 1 was used to help only show the Last Name of the Employee alongside the count of Order IDs associated with that last name.

c.
```
SELECT ProductName, ProductID, Quantity
FROM Customers
LEFT JOIN Orders USING (CustomerID)
LEFT JOIN OrderDetails USING (OrderID)
LEFT JOIN Products USING (ProductID)
WHERE Country = 'Germany'
ORDER BY Quantity DESC LIMIT 1;
```

FINAL ANSWER: Steeleye Stout and 35 is the Product Name and Product ID of the product that was ordered the most by customers in Germany

This solution takes into consideration the joining of four tables – Customers, Orders, OrderDetails and Products. A left join is made use of for each join. All tables were joined on the appropriate columns (CustomerID, OrderID, ProductID). A table where Country is Germany will be returned as this was stated in the WHERE clause of the above query. The resulting table was ordered by the Quantity of products obtained from the OrderDetails table. The order was done in a descending order to get the product with the highest orders. Futhermore, a limit of 1 was used to help only show the product that was ordered the most by customers in Germany.