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

Answer

a) *On observing the data given in the file, the existing formula to calculate the **Average Order Value** is based on the sum of all **order_amount** divided by **total number of orders i.e 5000**. This formula fails to take the number of items in consideration that are associated with each order ID and hence we get a bizarre value average order value of $3145.13*

K7 =SUM(D2:D5001)/COUNT(D2:D5001)

	order_id	shop_id	user_id	order_amount	total_items	payment_	created_at
2	1	53	746	224	2	cash	3/13/17 12:36
3	2	92	925	90	1	cash	3/3/17 17:38
4	3	44	861	144	1	cash	3/14/17 4:23
5	4	18	935	156	1	credit_card	3/26/17 12:43
6	5	18	883	156	1	credit_card	3/1/17 4:35
7	6	58	882	138	1	credit_card	3/14/17 15:25
8	7	87	915	149	1	cash	3/1/17 21:37
9	8	22	761	292	2	cash	3/8/17 2:05
10	9	64	914	266	2	debit	3/17/17 20:56
11	10	52	788	146	1	credit_card	3/30/17 21:08
12	11	66	848	322	2	credit_card	3/26/17 23:36
13	12	40	983	322	2	debit	3/12/17 17:58
14	13	54	799	266	2	credit_card	3/16/17 14:15
15	14	100	709	111	1	cash	3/22/17 2:39
16	15	87	849	447	3	credit_card	3/10/17 11:23
17	16	42	607	704000	2000	credit_card	3/7/17 4:00
18	17	17	731	176	1	cash	3/21/17 4:23
19	18	28	752	164	1	credit_card	3/21/17 12:09
20	19	83	761	258	2	cash	3/17/17 13:18
21	20	63	898	408	3	credit_card	3/29/17 15:11
22	21	66	987	322	2	cash	3/30/17 20:11
23	22	97	789	486	3	credit_card	3/4/17 15:44
24	23	88	985	704	4	credit_card	3/22/17 1:19
25	24	75	964	256	2	credit_card	3/12/17 3:07
26	25	73	917	495	3	cash	3/3/17 13:01
27	26	82	848	177	1	cash	3/25/17 21:35
28	27	47	882	145	1	cash	3/22/17 7:38
29	28	53	942	112	1	credit_card	3/17/17 9:41
30	29	40	944	322	2	cash	3/5/17 2:12
31	30	59	790	178	1	credit_card	3/4/17 22:49
32	31	76	857	310	2	cash	3/23/17 21:34
33	32	57	839	294	2	debit	3/19/17 5:31
34	33	76	712	465	3	credit_card	3/10/17 23:54
35	34	7	800	224	2	credit_card	3/3/17 5:31
36	35	34	704	244	2	debit	3/13/17 0:00

	J	K
7	AVG ORDER VALUE Current	3145.128

b) *The rectify the above, the correct **average order value** must be calculated as sum(order_amount)/sum(total_items)*

K7 =ROUND(SUM(D2:D5001)/SUM(E2:E5001),2)

	order_id	shop_id	user_id	order_amount	total_items	payment_m	created_at
2	1	53	746	224	2	cash	3/13/17 12:36
3	2	92	925	90	1	cash	3/3/17 17:38
4	3	44	861	144	1	cash	3/14/17 4:23
5	4	18	935	156	1	credit_card	3/26/17 12:43
6	5	18	883	156	1	credit_card	3/1/17 4:35
7	6	58	882	138	1	credit_card	3/14/17 15:25
8	7	87	915	149	1	cash	3/1/17 21:37
9	8	22	761	292	2	cash	3/8/17 2:05
10	9	64	914	266	2	debit	3/17/17 20:56
11	10	52	788	146	1	credit_card	3/30/17 21:08
12	11	66	848	322	2	credit_card	3/26/17 23:36
13	12	40	983	322	2	debit	3/12/17 17:58
14	13	54	799	266	2	credit_card	3/16/17 14:15
15	14	100	709	111	1	cash	3/22/17 2:39
16	15	87	849	447	3	credit_card	3/10/17 11:23
17	16	42	607	704000	2000	credit_card	3/7/17 4:00
18	17	17	731	176	1	cash	3/21/17 4:23
19	18	28	752	164	1	credit_card	3/21/17 12:09
20	19	83	761	258	2	cash	3/17/17 13:18
21	20	63	898	408	3	credit_card	3/29/17 15:11
22	21	66	987	322	2	cash	3/30/17 20:11
23	22	97	789	486	3	credit_card	3/4/17 15:44
24	23	88	985	704	4	credit_card	3/22/17 1:19
25	24	75	964	256	2	credit_card	3/12/17 3:07
26	25	73	917	495	3	cash	3/3/17 13:01
27	26	82	848	177	1	cash	3/25/17 21:35
28	27	47	882	145	1	cash	3/22/17 7:38
29	28	53	942	112	1	credit_card	3/17/17 9:41
30	29	40	944	322	2	cash	3/5/17 2:12
31	30	59	790	178	1	credit_card	3/4/17 22:49
32	31	76	857	310	2	cash	3/23/17 21:34
33	32	57	839	294	2	debit	3/19/17 5:31
34	33	76	712	465	3	credit_card	3/10/17 23:54
35	34	7	800	224	2	credit_card	3/3/17 5:31
36	35	34	704	244	2	debit	3/13/17 0:00

	J	K
7	ACTUAL AVG ORDER VALUE	$ 357.92

c) *The average order value for the month of March'17 is **$357.92***

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?

Answer

a) Here I have joined the shippers and orders tables on the common key shipper id.I extracted the shipper's name and took the count of unique order IDs .

```
SELECT shippername,
       Count(DISTINCT orderid) AS 'Packages Shipped'
FROM   orders o
       JOIN shippers s
         ON o.shipperid = s.shipperid
WHERE  shippername = 'Speedy Express'
```

Number of Records: 1

ShipperName	Packages Shipped
Speedy Express	54

b) Joined orders and employees table on employee id and extracted the last name of the person with the most number of orders placed.

```
SELECT e.lastname,
       Count(DISTINCT orderid) AS 'Total Orders Placed'
FROM   orders o
       JOIN employees e
         ON e.employeeid = o.employeeid
GROUP  BY 1
ORDER  BY 2 DESC
LIMIT  1
```

Number of Records: 1

lastname	Total Orders Placed
Peacock	40

c) For the most ordered product in Germany , I came up with 2 solutions . The first one is based on the quantity of the product , the second one is based on the number of times a product was ordered.

Solution 1 (Most Orders Based on Quantity)

```
SELECT c.country         AS OrderCountry,
       od.productid,
       p.productname,
       Sum(od.quantity) AS Total_Quantity
FROM   orderdetails od
       JOIN orders o
         ON od.orderid = o.orderid
       JOIN customers c
         ON o.customerid = c.customerid
       JOIN products p
         ON od.productid = p.productid
WHERE  country = 'Germany'
GROUP  BY 1,
          2
ORDER  BY total_quantity DESC
LIMIT  1
```

Number of Records: 1

OrderCountry	ProductID	ProductName	Total_Quantity
Germany	40	Boston Crab Meat	160

Solution 2 (Based on Number of Orders)

```
SELECT od.productid,
       p.productname,
       Count(od.productid) AS Times_Ordered,
       Sum(od.quantity)    AS Total_Order_Quantity
FROM   orderdetails od
       JOIN orders o
         ON od.orderid = o.orderid
       JOIN customers c
         ON o.customerid = c.customerid
       JOIN products p
         ON od.productid = p.productid
WHERE  country = 'Germany'
GROUP  BY od.productid,
          p.productname
ORDER  BY times_ordered DESC,
          total_order_quantity DESC
LIMIT  2
```

Number of Records: 2

ProductID	ProductName	Times_Ordered	Total_Order_Quantity
31	Gorgonzola Telino	5	125
40	Boston Crab Meat	4	160