***מסדי נתונים מטלה ראשונה מגישים מתן גרינברג וזיו מדר***

***206094815-311447783***

1. השאילתה שולפת שתי עמודות, הראשונה זה CustomerID אבל רק אלה שמתחילים ב -A והעמודה השנייה היא סכום כל ההזמנות של אותו לקוח כפול כמות של אותו מוצר, ולבסוף הטבלה מסודרת לפי המחיר של סכום כל המוצרים בסדר יורד.

בפירוט: בהתחלה בוחרים את העמודה CustomerID ו – price בתור סכום המוצרים, מתוך הטבלה החדשה שיוצרים שהיא orders מחוברת עם order details אבל רק אם ה idשל ההזמנה תואם כאשר ה CustomerID מתחיל ב – A ומסודר בסדר יורד לפי ה price .

הטבלה הסופית:

|  |  |  |
| --- | --- | --- |
| price | CustomerID |  |
| 13806.5000 | AROUT |  |
| 7515.3500 | ANTON |  |
| 4596.2000 | ALFKI |  |
| 1402.9500 | ANATR |  |

1. Select OrderId from customers as C join employees as E join orders as O

on C.city = E.city and E.EmployeeID = O.EmployeeID

and C.CustomersID = O.CustomersId

order by OrderID desc

הטבלה הסופית:

'11047'

'11016'

'10953'

'10869'

'10848'

'10804'

'10696'

'10599'

'10596'

'10539'

'10538'

'10532'

'10523'

'10469'

'10359'

'10355'

'10289'

1. Create temporary table group\_categories

select CategoryID as ID, count (CategoryID) as Pdct,

avg(UnitPrice) as average from products group by CategoryID;

select Pdct as NUM, count(Pdct) as COUNT,

avg(average) as average

from group\_categories group by Pdct;

הטבלה הזמנית:

ID Pdct average

'1', '12', '37.97916667'

'2', '12', '23.06250000'

'3', '13', '25.16000000'

'4', '10', '28.73000000'

'5', '7', '20.25000000'

'6', '6', '54.00666667'

'7', '5', '32.37000000'

'8', '12', '20.68250000'

הטבלה הסופית:

NUM COUNT average

'12', '3', '27.241388890000'

'13', '1', '25.160000000000'

'10', '1', '28.730000000000'

'7', '1', '20.250000000000'

'6', '1', '54.006666670000'

'5', '1', '32.370000000000'

1. select CategoryID, count(CategoryID) as sum from customers join orders join `order details` join products

where ContactName like '%sommer' and `order details`.`ProductID` = products.ProductID

and orders.OrderID = `order details`.`OrderID`

and customers.CustomerID = orders.CustomerID group by CategoryID order by CategoryID;

טבלה סופית:

ID sum

'1', '1'

'2', '1'

'5', '2'

'6', '2'

1. create temporary table dates

select OrderDate from orders

where EmployeeID = 8 and OrderDate >= date\_add('1998-05-08 00:00:00', interval - 14 day);

select orders.EmployeeID,LastName,FirstName, max(date(orders.OrderDate)) as max\_date,

date\_add( max(date(orders.OrderDate)), interval 14 day) as release\_date

from orders join dates join employees

where orders.OrderDate = dates.OrderDate and orders.EmployeeID <> 8 and orders.EmployeeID = employees.EmployeeID

group by orders.EmployeeID

order by orders.EmployeeID;

יצרנו טבלה חדשה בשם – dates שהיא כל התאריכים שבועיים לפני שגילו ש – 8 חולה (לקחנו שבועיים אחורה כי יכול להיות שהוא נדבק גם לפני), וצירפנו אותה לטבלה המקורית אבל רק בתאריכים ש – 8 עבד, משם שלפנו את התאריך הכי גדול החופף והוספנו עמודה של בידוד לשבועיים אחרי.

טבלה סופית:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Release\_date | Max\_date | FirstName | LastName | EmployeeID |
| 1998-05-20 | 1998-05-06 | Nancy | Davolio | 1 |
| 1998-05-11 | 1998-04-27 | Andrew | Fuller | 2 |
| 1998-05-11 | 1998-04-27 | Janet | Leverling | 3 |
| 1998-05-20 | 1998-05-06 | Margaret | Peacock | 4 |
| 1998-05-20 | 1998-05-06 | Robert | King | 7 |