



```
—Q3
select *
from
  (SELECT e.EmployeeID,
    FirstName + ' ' + LastName FullName,
    sum(Quantity * UnitPrice) sumRev,
    ROW_NUMBER() over(Order by sum(Quantity * UnitPrice)) rowNum,
    FIRST_VALUE(o.OrderID) over(partition by e.EmployeeID
                                order by sum(Quantity * UnitPrice) desc, o.OrderID) LargestOrderValue,
    FIRST_VALUE(sum(Quantity * UnitPrice)) over(partition by e.EmployeeID
                                                order by sum(Quantity * UnitPrice) desc) LargestSumRev,
    FIRST_VALUE(OrderDate) over(partition by e.EmployeeID
                                order by sum(Quantity * UnitPrice) desc, OrderDate) EarliestOrderDate
  from Employees e join Orders o
    on e.EmployeeID = o.EmployeeID
   join [Order Details] od
    on o.OrderID = od.OrderID
   where OrderDate between '1996-07-30' and '1997-07-30'
   group by e.EmployeeID, FirstName, LastName, o.OrderID, OrderDate) Q1
  where rowNum = 1
  order by rowNum
```

### שאלה 3

- מס' הלקוח.
  - שם מלא של הלקוח.
  - סך ה~~פדיון~~<sup>המכירות</sup> של העובד לשנה האחרונות.
  - דירוג העובד בהתאם על סך ה~~פדיון~~<sup>מכירותו</sup> לשנה האחרונות.
  - מס' הזמן שבעל סך ה~~פדיון~~<sup>הגבוה ביותר</sup> של אותו עובד לשנה האחרונות.
  - סך ה~~פדיון~~<sup>המכירות</sup> של ההזמנה*הכי גדולה* לשנה האחרונות.
  - תאריך ההזמנה הראשון של ההזמנה*הכי גדולה* של העובד.

הזמןה הכי גדולה = הזמןה שסר הפדיון שלה הוא הגבוה ביותר.  
על מנת שהזמןה לא יჩזרו מספר פעמים בדרכו יש להשתמש בפונקציית Row Number.

## שאלה 4

צרו פרוצדורה בשם GetProductSalesSummary המקבלת 2 משתנים:

- תאריך התחלת מסוג Date
  - תאריך סיום מסוג Date

על הפורצדורה להחזיר סיכון מכירות של כל המוצרים בטוויה התאריכים שבחר.

- לכל מוצר יש להציג:

  1. מספר המוצר
  2. שם המוצר – *prod ID*
  3. סך הכמות (quantity) שנקרכו ממנו
  4. סך הפדיון מההכנסות מכירות המוצר
  5. דירוג המוצר בהתאם על סך הפדיון מהמכירות ממנו.
  6. ממוצע הפדיון מכירות המוצר.

orders ← יש לבדוק את הפרטציה על כל שנה מ-1997

```

SELECT Country,
       count(distinct c.CustomerID) countCust,
       COUNT(DISTINCT o.OrderID) countOrders,
       avg(Quantity) avgQuan,
       count(DISTINCT od.ProductID) countProds
  from Customers c join Orders o
    on c.CustomerID = o.CustomerID
   join [Order Details] od
    on o.OrderID = od.OrderID
group by Country
having COUNT(DISTINCT o.OrderID) >
           (select count(DISTINCT o.OrderID)
              from Orders o join [Order Details] od
                on o.OrderID = od.OrderID
               where CustomerID like 'quick')
           (select count(DISTINCT o.OrderID)
              from Orders o join [Order Details] od
                on o.OrderID = od.OrderID
               where CustomerID like 'alfk1') |

```

Sum(  $\rightarrow$  ~~desr~~  $\Rightarrow$   $\lambda x. \text{max}(\text{f}, \text{g})$  )  $\in$   $\text{L}_\text{HOL}$

רכל מדריכים  
מנהלת מחלקה תושבי חוץ מעוניינת במידע על מדינות לצורך בחינת הלקחות

- לשם כך, יש להציג את הדוח הבא:

א. שם המדיינָה  
ב. כמות הלקוחות במדינָה

הדו"ח מבוסס מידע הכלול אף ורק במקרים בהם כמות ההזמנות של הלוקחות, גודלה מההפרש בין כמות ההזמנות שבייצ' הלקוח QUICK לבין כמות

## הזמןות שביצע הלקוח ALFKI

distinct elements in the set of countable sets of natural numbers.

- (1) **group by** **רף עליה על המאפיין הקיים, רף עליה**  
 case when when the feature exists, when it does not

**תיאורטיות 1**  
בBOR מנהלי מערכות המידע של מכון אימונים מוביל, התקשרות לסטודנטים לפי מדרג רישום המבוסס על כמות הביקורים שלהם בחדר הcorsה. בחרבו גיברילוביטוב רגולטונו.

- פלטינום: כמות ביקורם השווה או הגדולה מ-100%
  - זהב: כמות ביקורם בין 50 ל-99%
  - כסף: כמות ביקורם בין 25 ל-49%
  - ברונזה: כמות ביקורם הנמוכה מ-25%

הסבירו שלב אחריו שלב כיצד תבצעו את סיווג המתאימים לקריטריונים השונים באמצעות SQL ללא כתיבת קוד.

## תיאורטית 2

המצביע על מילוי הדרישות. בפועל מושג המצביע על מילוי הדרישות על ידי ביצוע הדרישות. בפועל מושג המצביע על מילוי הדרישות על ידי ביצוע הדרישות.

אחריה: מומר להניח הנקודות לאבי טבלאות שטמודות. רק יש לציין את הנקודות.

טיורטית 3

בBOR אנדלייסטים תוחחים בחברת eCommerce, קיבלתם מシימת חינכה לעובד חדש שהגיעו ואותם רוצים לחתן לו פרויקט שקיבלותם ממנהלת שימור לקוחות. המנהלת רצתה לעקוב אחר שיעורי שימור המשתמשים לאורוך זמן ולזהות מגמות בעורבות משתמשים.

על מנת לנתח מגמות בשימור משתמשים ומעורבותם יש לבצע השוואות יחס בין משתמשים חדשים למשתמשים פעילים על פci תקופות זמן עוקבות. הסבירו לעובד החוטר שאתם חונכין כיצד יש לבצע את השאלה הנ"ל ע"י שימוש בפונקציות חלון שלב אחר שלב ללא כתיבת קוד.

הארה: מותר להניח הנחות. לציין כאשר ביצעתם זאת.

## תיאורטיות 4

מה הבדל בין כת שאלתה ב- (Derived Tables) FROM לtáבלאות זמניות ?(temp tables)  
יש לנו מנו לפחות 3 הבדלים.

స్వయం తను ఉచ్చి CTE, అందుల్లోనికి ఉన్న కారణం from a surface ను  
from a surface ను, ఏ నుండి ఆమె గిరి CTE కి ఉన్న

תיאורטית 5

להלן קטע קוד:

Digitized by srujanika@gmail.com

להן קטע קוד:  
בז' יוניסטרטיבי SQL  
לפנינו מופיע שאלת חיפוש על כל הזמנות ששלחו לארצות הברית.  
השאלה נראית כזאת:

```
select *  
from Orders as o  
where o.ShipCountry like 'usa' and  
o.OrderID in  
{  
    (SELECT od.OrderID  
     from [Order Details] as od  
     where od.ProductID in (SELECT od.ProductID  
                           from [Order Details] as od  
                           where od.OrderID in (select o.OrderID  
                                     from orders as o  
                                     where o.CustomerID like 'quick')))  
}
```

השאלה מוגדרת באמצעות סעיפים (Nested Subqueries).  
הSEQUENCE 1 (הSEQUENCE החיצוני ביותר):  
SELECT od.OrderID  
from [Order Details] as od  
where od.ProductID in (SEQUENCE 2)  
SEQUENCE 2:  
SELECT od.ProductID  
from [Order Details] as od  
where od.OrderID in (SEQUENCE 3)  
SEQUENCE 3:  
select o.OrderID  
from orders as o  
where o.CustomerID like 'quick'  
הSEQUENCE 3 הוא סעיף פנימי של סעיף פנימי.

רשום את השאלה עבורה בוצע הפתרון.