

Tracking Running Totals with Window Functions

Business problem: Tracking Sales Quota Progress over Time

The sales team works diligently to sell the product, and they have quotas that they must reach in order to earn all of their commission. Because these goals are so intimately tied to revenue, the manager of the team wants to track each sales member's performance throughout the year. You suggest a **% of quota reached** metric that could be displayed on a dashboard, but the sales manager expresses her concern that a single metric won't give her visibility into their progress throughout the year. You suggest providing a **running_total** of sales revenue and a **percent_quota** metric that will be recalculated every time a sales member makes another sale. She agrees, and you get started!

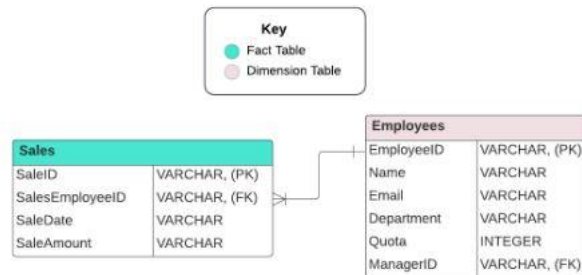
Task:

Calculate the running total of sales revenue, **running_total**, and the % of quota reached, **percent_quota**, for each sales employee on each date they make a sale. Use the **sales** and **employees** table to pull in and create the following fields:

- **salesemployeeid**
- **saledate**
- **saleamount**
- **quota**
- **running_total**
- **percent_quota**

Order the final output by **salesemployeeid** and **saledate**.

Sales Data Model



```
1  -- SQL code by John Uzoma
2  SELECT
3      S.salesemployeeid,
4      S.saledate,
5      S.saleamount,
6      SUM(S.saleamount) OVER(PARTITION BY S.salesemployeeid ORDER BY S.saledate) AS running_total,
7      CONCAT(
8          SUM(S.saleamount) OVER(PARTITION BY S.salesemployeeid ORDER BY S.saledate) * 100 / E.quota,
9          '%'
10     ) as percent_quota
11 FROM SALES S
12 INNER JOIN EMPLOYEES E
13 on S.salesemployeeid = E.employeeid
14 ORDER BY S.salesemployeeid, S.saledate;
15
```

Result

SALESEMPLOYEEID	SALEDATE	SALEAMOUNT	RUNNING_TOTAL	PERCENT_QUOTA
E172	2023-03-11	9000	9000	18%
E172	2023-06-11	11000	20000	40%
E172	2023-06-15	15000	35000	70%
E172	2023-07-17	3000	38000	76%
E192	2023-02-09	14000	14000	23%
E192	2023-03-14	9000	23000	38%
E192	2023-05-16	16000	39000	65%
E429	2023-01-30	4000	4000	5%
E429	2023-04-03	7000	11000	13%
E429	2023-07-13	10000	21000	26%
E738	2023-01-04	18000	18000	36%
E738	2023-03-18	10000	28000	56%
E738	2023-03-26	5000	33000	66%
E738	2023-04-16	1000	34000	68%
E738	2023-07-16	12000	46000	92%
E738	2023-07-28	11000	57000	114%