Question:

Sometimes products for whatever reason stop selling and a symptom can be an item that was selling well faces a stock out or delisting (or something else). Write a query that shows products that have sold for more than 30 days in the last 60 days, but hasn't had sales for the last week.

You may assume a sales table schema of your preference.

1. Date
2. Product_id
3. Total Items sold to date
4. number of days with sales
5. number of dates in the recent history where sales have ceased

Assuming the following table:

```
CREATE table DWD_ORDERS (
  DATE DATE,
  PRODUCT_ID VARCHAR NOT NULL,
  TOTAL_ITEMS_SOLD_TO_DATE LONG,
  NUMBER_OF_DAYS_WITH_SALES INT,
  NUMBER_OF_DATES_WHERE_SALES_CEASED INT,
);
```

Assuming Snowflake query syntax, and assuming that NUMBER_OF_DAYS_WITH_SALES and NUMBER_OF_DATES_WHERE_SALES_CEASED are historical cumulative data.

Suppose the day of interest is 2023-06-01

```
SELECT CURR.PRODUCT_ID
FROM (
```

```
        SELECT PRODUCT_ID
                ,NUMBER_OF_DAYS_WITH_SALES
        FROM DWD_ORDERS
        WHERE DATE = '2023-06-01'
        ) CURR
INNER JOIN (
        SELECT PRODUCT_ID
                ,NUMBER_OF_DAYS_WITH_SALES
        FROM DWD_ORDERS
        WHERE DATE = DATEADD('DAY', - 7, '2023-06-01')
        ) ON LAST_WEEK.PRODUCT_ID = CURR.PRODUCT_ID
        AND LAST_WEEK.NUMBER_OF_DAYS_WITH_SALES =
CURR.NUMBER_OF_DAYS_WITH_SALES
INNER JOIN (
        SELECT PRODUCT_ID
                ,NUMBER_OF_DAYS_WITH_SALESFROM DWD_ORDERS
        WHERE DATE = DATEADD('DAY', - 60, '2023-06-01')
        ) PREV_60D ON CURR.PRODUCT_ID = PREV_60D.PRODUCT_ID
        AND CURR.NUMBER_OF_DAYS_WITH_SALESFROM -
PREV_60D.NUMBER_OF_DAYS_WITH_SALESFROM > 30;
```

The next part of the question is how do we actually build this table:

```
CREATE table DWD_ORDERS (
  DATE DATE,
  PRODUCT_ID VARCHAR NOT NULL,
  TOTAL_ITEMS_SOLD_TO_DATE LONG,
  NUMBER_OF_DAYS_WITH_SALES INT,
  NUMBER_OF_DATES_WHERE_SALES_CEASED INT,
);
```

Without using filters and group by.

For such cases, we assume that there is a fact table containing the following as per q1:

```
CREATE table dwd_orders (
  QUANTITY INT NOT NULL,
  PRODUCT_ID VARCHAR NOT NULL,
  PARTITION_DATE DATE
);
```

We can use window function to do what we want

```
Select orders.PARTITION_DATE, orders.PRODUCT_ID
, orders.total_number_of_items_sold_to_date,
SUM(DAYS_WITH_SALES ) OVER (PARTITION BY PRODUCT_ID
                    ORDER BY PARTITION_DATE  asc) AS
number_of_days_with_sales,

SUM(CASE WHEN DAYS_WITH_SALES = 1 THEN 0 ELSE 1 END ) OVER
(PARTITION BY PRODUCT_ID
                    ORDER BY PARTITION_DATE  asc) AS
number_of_days_without_sales

FROM (select PARTITION_DATE, PRODUCT_ID,
SUM(QUANTITY) OVER (PARTITION BY PRODUCT_ID
                    ORDER BY PARTITION_DATE  asc) AS
total_number_of_items_sold_to_date,

COUNT(DISTINCT case when QUANTITY > 0 then 1 else NULL end) OVER
(PARTITION BY PRODUCT_ID) AS DAYS_WITH_SALES from dwd_orders

) orders
```