SELECT

transaction\_id,

transaction\_date,

transaction\_time,

transaction\_qty,

store\_id,

store\_location,

product\_id,

unit\_price,

product\_category,

product\_type,

product\_detail

FROM

"COFFEE"."SHOP"."TRANSACTIONS";

-- Monthly sales analysis:

SELECT

TO\_CHAR(transaction\_date, 'YYYY-MM') AS Month,

SUM(unit\_price \* transaction\_qty) AS total\_sales,

COUNT(\*) AS number\_of\_transactions,

AVG(unit\_price \* transaction\_qty) AS average\_transactions\_value

FROM

"COFFEE"."SHOP"."TRANSACTIONS"

GROUP BY

TO\_CHAR(transaction\_date, 'YYYY-MM')

ORDER BY

Month ASC;

-- track sales performance across multiple locations

SELECT

store\_location,

TO\_CHAR(transaction\_date, 'YYYY-MM') AS Month,

SUM(unit\_price \* transaction\_qty) AS total\_sales,

COUNT(\*) AS number\_of\_transactions,

AVG(unit\_price \* transaction\_qty) AS average\_transactions\_value

FROM

"COFFEE"."SHOP"."TRANSACTIONS"

GROUP BY

store\_location,

TO\_CHAR(transaction\_date, 'YYYY-MM')

ORDER BY

store\_location,

TO\_CHAR(transaction\_date, 'YYYY-MM');

-- Calculation for total revenue per product

SELECT

product\_type,

SUM(unit\_price \* transaction\_qty) AS total\_revenue

FROM

transactions

GROUP BY

product\_type

ORDER BY

total\_revenue DESC;

-- To determine the quantity of items sold by product\_category

SELECT

product\_category,

SUM(transaction\_qty) AS total\_quantity\_sold

FROM

transactions

GROUP BY

product\_category

ORDER BY

total\_quantity\_sold DESC;

--to determine the best-selling product types or details

SELECT

store\_location,

product\_type,

product\_detail,

SUM(unit\_price \* transaction\_qty) AS total\_revenue

FROM

transactions

GROUP BY

store\_location,

product\_type,

product\_detail

ORDER BY

SUM(transaction\_qty) DESC,

total\_revenue DESC

LIMIT 10;

--products with low total sales revenue and quantity over a specified period

WITH product\_sales AS (

SELECT

product\_id,

product\_detail,

SUM(transaction\_qty) AS total\_quantity,

SUM(unit\_price \* transaction\_qty) AS total\_revenue

FROM

transactions

WHERE

transaction\_date BETWEEN '2024-01-01' AND '2024-06-30'

GROUP BY

product\_id,

product\_detail

)

SELECT

product\_id,

product\_detail,

total\_quantity,

total\_revenue

FROM

product\_sales

WHERE

total\_quantity < 50

AND total\_revenue < 500

ORDER BY

total\_revenue ASC;

--identifies which time of the day your store performs best:

SELECT

store\_location,

hour\_of\_day,

total\_revenue

FROM (

SELECT

store\_location,

EXTRACT(HOUR FROM transaction\_time) AS hour\_of\_day,

SUM(unit\_price \* transaction\_qty) AS total\_revenue,

ROW\_NUMBER() OVER (PARTITION BY store\_location ORDER BY SUM(unit\_price \* transaction\_qty) DESC) AS rn

FROM

transactions

GROUP BY

store\_location,

hour\_of\_day

)

WHERE rn = 1;

--underperforming products

SELECT

store\_location,

LISTAGG(product\_detail, ', ') WITHIN GROUP (ORDER BY total\_revenue ASC) AS underperforming\_products,

MIN(total\_revenue) AS lowest\_revenue

FROM (

SELECT

store\_location,

product\_id,

product\_detail,

SUM(transaction\_qty) AS total\_quantity,

SUM(unit\_price \* transaction\_qty) AS total\_revenue,

ROW\_NUMBER() OVER (

PARTITION BY store\_location

ORDER BY SUM(unit\_price \* transaction\_qty) ASC

) AS rn

FROM

transactions

GROUP BY

store\_location,

product\_id,

product\_detail

)

WHERE rn = 1

GROUP BY store\_location

ORDER BY store\_location;