SELECT

\*

FROM

"COFFEE1"."SHOPS"."TRANSACTIONS";

-- 1. Sales by Product Category and 30-Minute Time Intervals

SELECT

-- Format date as 'YYYY-MM-DD'

TO\_CHAR(TRANSACTION\_TIME, 'YYYY-MM-DD') AS sales\_date,

-- Extract hour and calculate 30-minute interval

TO\_CHAR(TRANSACTION\_TIME, 'HH24') || ':' || LPAD(CAST(FLOOR(EXTRACT(MINUTE FROM TRANSACTION\_TIME) / 30) \* 30 AS VARCHAR), 2, '0') AS sales\_30min\_interval,

-- Product category

PRODUCT\_CATEGORY,

-- Total sales calculation

SUM(Unit\_PRICE \* TRANSACTION\_QTY) AS total\_sales

FROM

TRANSACTIONS

WHERE

PRODUCT\_CATEGORY IS NOT NULL

AND TRANSACTION\_TIME IS NOT NULL

GROUP BY

sales\_date,

sales\_30min\_interval,

PRODUCT\_CATEGORY

ORDER BY

sales\_date,

sales\_30min\_interval,

PRODUCT\_CATEGORY;

-- 2. High-Performing Products (Top 5 by Total Revenue)

SELECT

product\_type,

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

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

FROM

transactions

WHERE

product\_type IS NOT NULL

GROUP BY

product\_type

ORDER BY

total\_revenue DESC

LIMIT 5;

-- 3. Low-Performing Products (Bottom 5 by Total Revenue, excluding those with no sales)

SELECT

product\_type,

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

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

FROM

transactions

WHERE

product\_type IS NOT NULL

GROUP BY

product\_type

HAVING

SUM(unit\_price \* transaction\_qty) > 0

ORDER BY

total\_revenue ASC

LIMIT 5;

-- 4. Total Revenue for Each Day

SELECT

transaction\_date,

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

FROM

Transactions

GROUP BY

transaction\_date

ORDER BY

transaction\_date;

SELECT

TO\_CHAR(transaction\_date, 'YYYY-MM') AS sales\_month, -- Extracts Year and Month

product\_detail, -- Identifies the specific product

SUM(transaction\_qty \* unit\_price) AS monthly\_revenue -- Calculates total revenue for the product in that month

FROM

Transactions

WHERE

transaction\_date IS NOT NULL

AND product\_detail IS NOT NULL

AND transaction\_qty IS NOT NULL

AND unit\_price IS NOT NULL

GROUP BY

sales\_month,

product\_detail

ORDER BY

sales\_month ASC,

monthly\_revenue DESC;

SELECT

store\_location,

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

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

COUNT(DISTINCT transaction\_id) AS total\_transactions,

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

FROM

Transactions

WHERE

store\_location IS NOT NULL

AND transaction\_qty IS NOT NULL

AND unit\_price IS NOT NULL

AND transaction\_id IS NOT NULL

GROUP BY

store\_location

ORDER BY

total\_revenue DESC;