SQL ASSIGNMENT 03

PROBLEM STATEMENT

1. Retrieve the total sales quantity and revenue for each product.

2. Find the total revenue for each customer.

3. Get the products with more than 10 units sold in a single order.

4. List the customers who have placed orders on at least three different dates.

5. Calculate the average unit price of products

. 6. Find products with an average unit price greater than $12.00.

7. Retrieve the customers who have spent more than $100.00 in total.

8. List the customers who have purchased 'Widget B' and 'Widget A' in the same order.

DATABASE

Created a Demodatabase in the Snowflake and then run a command **Use Database**

```
USE DATABASE DEMODATABASE;
```

Creating a **Sales Table**

```
CREATE OR REPLACE TABLE SALES
 (
 ORDER_ID INT PRIMARY KEY,
 CUSTOMER_ID INT,
 PRODUCT_ID INT,
 PRODUCT_NAME VARCHAR(50),
 QUANTITY INT,
 UNIT_PRICE DECIMAL(10, 2),
 ORDER_DATE DATE
 );
```

-- INSERTING DATA INTO THE **Sales Table**

```
 --INSERTING THE DATA INTO THE TABLE
INSERT INTO sales (order_id, customer_id, product_id, product_name, quantity, unit_price, order_date)
VALUES
 (1, 101, 1, 'Widget A', 5, 10.00, '2023-01-15'),
 (2, 102, 2, 'Widget B', 2, 12.50, '2023-01-16'),
 (3, 103, 1, 'Widget A', 3, 10.00, '2023-01-16'),
 (4, 104, 3, 'Widget C', 1, 15.75, '2023-01-17'),
 (5, 105, 2, 'Widget B', 4, 12.50, '2023-01-17'),
 (6, 106, 1, 'Widget A', 2, 10.00, '2023-01-18'),
 (7, 107, 4, 'Widget D', 3, 20.00, '2023-01-18'),
 (8, 108, 2, 'Widget B', 5, 12.50, '2023-01-19'),
 (9, 109, 1, 'Widget A', 1, 10.00, '2023-01-19'),
 (10, 101, 3, 'Widget C', 2, 15.75, '2023-01-20');
```

Q1. Retrieve the total sales quantity and revenue for each product.

```
SELECT PRODUCT_NAME,SUM(QUANTITY) AS TOTAL_SALES
,SUM(UNIT_PRICE * QUANTITY) AS REVENUE
FROM SALES
GROUP BY PRODUCT_NAME
ORDER BY REVENUE DESC ;
```

OUTPUT

	PRODUCT_NAME	⋯	TOTAL_SALES	REVENUE
1	Widget B		11	137.50
2	Widget A		11	110.00
3	Widget D		3	60.00
4	Widget C		3	47.25

Q2. Find the total revenue for each customer.

```
SELECT CUSTOMER_ID,
SUM(UNIT_PRICE * QUANTITY) AS REVENUE
FROM SALES
GROUP BY CUSTOMER_ID
ORDER BY REVENUE DESC;
```

OUTPUT

	CUSTOMER_ID	···	REVENUE
1	101		81.50
2	108		62.50
3	107		60.00
4	105		50.00
5	103		30.00
6	102		25.00
7	106		20.00
8	104		15.75
9	109		10.00

Q3. Get the products with more than 10 units sold in a single order.

```
SELECT PRODUCT_NAME, SUM(QUANTITY) AS TOTAL_QUANTITY
FROM SALES
GROUP BY PRODUCT_NAME
HAVING TOTAL_QUANTITY > 10;
```

OUTPUT

	PRODUCT_NAME	...	TOTAL_QUANTITY
1	Widget A		11
2	Widget B		11

Q4. List the customers who have placed orders on at least three different dates.

```
SELECT CUSTOMER_ID,COUNT(DISTINCT ORDER_DATE) AS ORDER_DATES
FROM SALES
GROUP BY 1
HAVING ORDER_DATES >=3 ;
```

OUTPUT

	CUSTOMER_ID	ORDER_DATES
Query produced no results		

Q5. Calculate the average unit price of products.

```
SELECT PRODUCT_NAME,ROUND(AVG(UNIT_PRICE),0) AS AVERAGE_UNIT_PRICE
FROM SALES
GROUP BY PRODUCT_NAME ;
```

OUTPUT

	PRODUCT_NAME	···	AVERAGE_UNIT_PRICE
1	Widget A		10
2	Widget B		13
3	Widget C		16
4	Widget D		20

Q6. Find the products with an average unit price greater than $12.00.

```
SELECT PRODUCT_NAME,ROUND(AVG(UNIT_PRICE),0) AS AVERAGE_UNIT_PRICE
FROM SALES
GROUP BY PRODUCT_NAME
HAVING AVERAGE_UNIT_PRICE >12;
```

OUTPUT

	PRODUCT_NAME	···	AVERAGE_UNIT_PRICE
1	Widget B		13
2	Widget C		16
3	Widget D		20

Q7. Retrieve the customers who have spent more than $100.00 in total.

```
SELECT CUSTOMER_ID, SUM(UNIT_PRICE * QUANTITY) AS TOTAL_SALES
FROM SALES
GROUP BY CUSTOMER_ID
HAVING TOTAL_SALES >100
ORDER BY TOTAL_SALES DESC;
```

OUTPUT

	CUSTOMER_ID	TOTAL_SALES
Query produced no results		

Q8. List the customers who have purchased 'Widget B' and 'Widget A' in the same order.

```
SELECT CUSTOMER_ID,COUNT(ORDER_ID) AS TOTAL_ORDERS
FROM SALES
WHERE PRODUCT_NAME = 'Widget B' AND PRODUCT_NAME = 'Widget A'
GROUP BY CUSTOMER_ID
HAVING TOTAL_ORDERS >2 ;
```

OUTPUT

	CUSTOMER_ID	TOTAL_ORDERS
Query produced no results		