

ZUDIO SALES SQL QUERIES

A. KPI's

We need to analyze key indicators for our Zudio sales data to gain insights into our business performance. Specifically, we want to calculate the following metrics:

1. Total Sales Revenue :

```
SELECT SUM(Price) AS Total_revenue  
FROM Zudio_sales_data;
```

The screenshot shows a SQL query editor with the query: `SELECT SUM(Price) AS Total_revenue FROM Zudio_sales_data;`. Below the editor, the 'Results' tab is active, displaying a table with one row and one column. The column is labeled 'Total_revenue' and the value is 13911035. The interface includes a toolbar with icons for undo, redo, and other functions, as well as a status bar showing 2 errors and 0 warnings.

| | Total_revenue |
|---|---------------|
| 1 | 13911035 |

2. Total Quantity Sold :

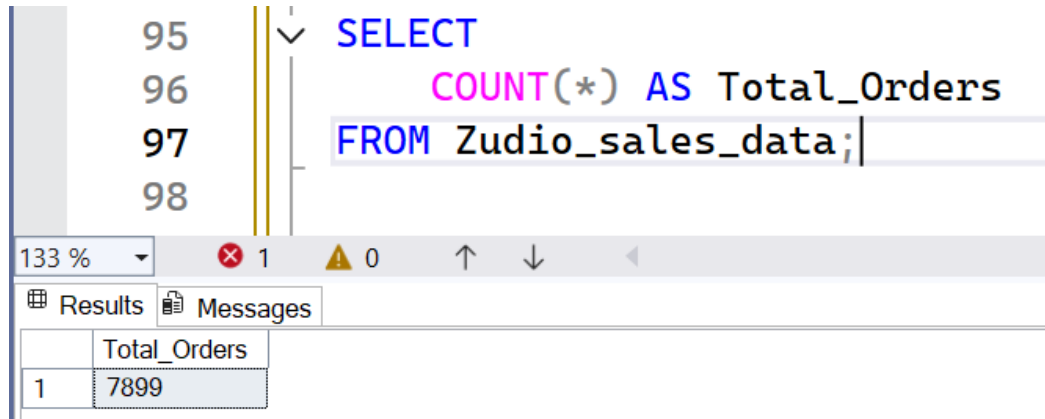
```
SELECT SUM(Quantity) AS Total_quantity  
FROM Zudio_sales_data;
```

The screenshot shows a SQL query editor with the query: `SELECT SUM(Quantity) AS Total_quantity FROM Zudio_sales_data;`. Below the editor, the 'Results' tab is active, displaying a table with one row and one column. The column is labeled 'Total_quantity' and the value is 35699. The interface includes a toolbar with icons for undo, redo, and other functions, as well as a status bar showing 2 errors and 0 warnings.

| | Total_quantity |
|---|----------------|
| 1 | 35699 |

3. Total Number of Orders :

```
SELECT  
    COUNT(*) AS Total_Orders  
FROM Zudio_sales_data;
```

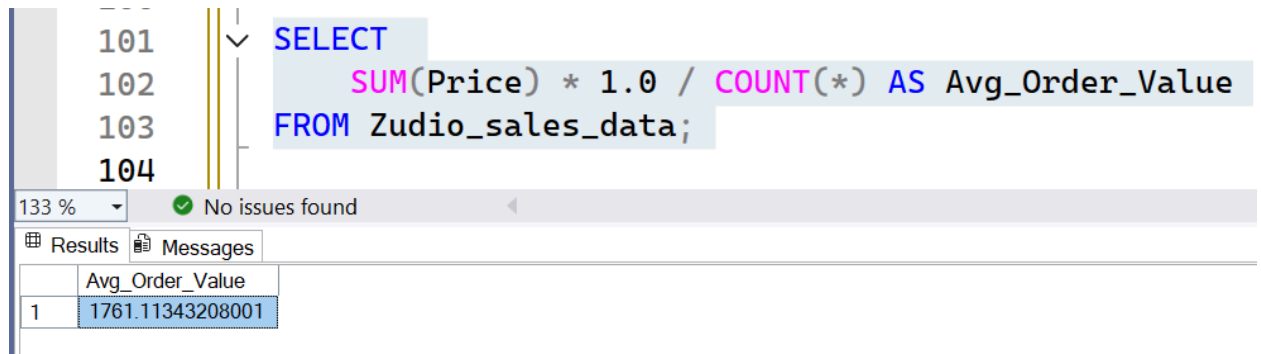


```
SELECT  
    COUNT(*) AS Total_Orders  
FROM Zudio_sales_data;
```

| Total_Orders |
|--------------|
| 7899 |

4. Average Order Value (AOV) :

```
SELECT  
    SUM(Price) * 1.0 / COUNT(*) AS Avg_Order_Value  
FROM Zudio_sales_data;
```

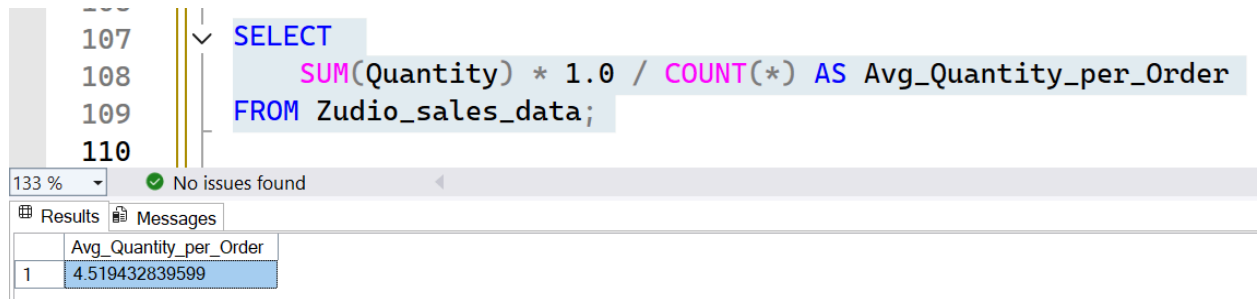


```
SELECT  
    SUM(Price) * 1.0 / COUNT(*) AS Avg_Order_Value  
FROM Zudio_sales_data;
```

| Avg_Order_Value |
|------------------|
| 1761.11343208001 |

5. Average Quantity per Order :

```
SELECT
    SUM(Quantity) * 1.0 / COUNT(*) AS Avg_Quantity_per_Order
FROM Zudio_sales_data;
```



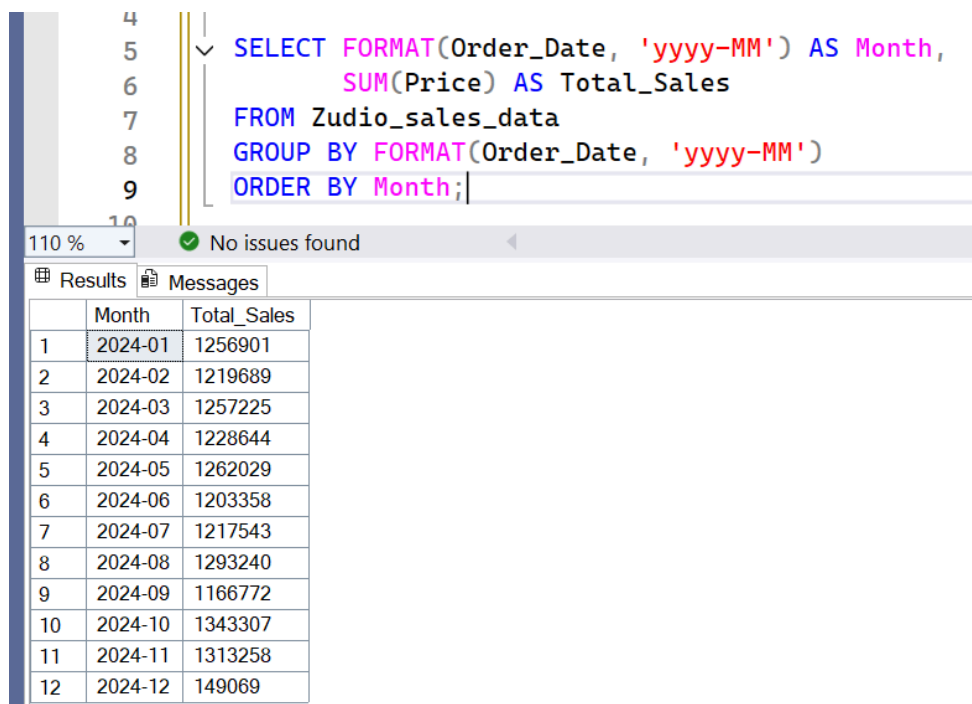
```
SELECT
    SUM(Quantity) * 1.0 / COUNT(*) AS Avg_Quantity_per_Order
FROM Zudio_sales_data;
```

| Avg_Quantity_per_Order |
|------------------------|
| 4.519432839599 |

B. PIVOT TABLES AND PIVOT CHARTS QUERIES

1. Total Sales Trend (Monthly)

```
SELECT FORMAT(Order_Date, 'yyyy-MM') AS Month,
       SUM(Price) AS Total_Sales
FROM Zudio_sales_data
GROUP BY FORMAT(Order_Date, 'yyyy-MM')
ORDER BY Month;
```



```
SELECT FORMAT(Order_Date, 'yyyy-MM') AS Month,
       SUM(Price) AS Total_Sales
FROM Zudio_sales_data
GROUP BY FORMAT(Order_Date, 'yyyy-MM')
ORDER BY Month;
```

| | Month | Total_Sales |
|----|---------|-------------|
| 1 | 2024-01 | 1256901 |
| 2 | 2024-02 | 1219689 |
| 3 | 2024-03 | 1257225 |
| 4 | 2024-04 | 1228644 |
| 5 | 2024-05 | 1262029 |
| 6 | 2024-06 | 1203358 |
| 7 | 2024-07 | 1217543 |
| 8 | 2024-08 | 1293240 |
| 9 | 2024-09 | 1166772 |
| 10 | 2024-10 | 1343307 |
| 11 | 2024-11 | 1313258 |
| 12 | 2024-12 | 149069 |

2. Sales by Product Category

```
SELECT
    Category,
    SUM(Quantity) AS Total_Qty_Sold,
    SUM(Price) AS Total_Sales
FROM Zudio_sales_data
GROUP BY Category
ORDER BY Total_Sales DESC;
```

The screenshot shows a SQL IDE interface. The query editor on the left contains the following SQL code:

```
13 SELECT
14     Category,
15     SUM(Quantity) AS Total_Qty_Sold,
16     SUM(Price) AS Total_Sales
17 FROM Zudio_sales_data
18 GROUP BY Category
19 ORDER BY Total_Sales DESC;
20
```

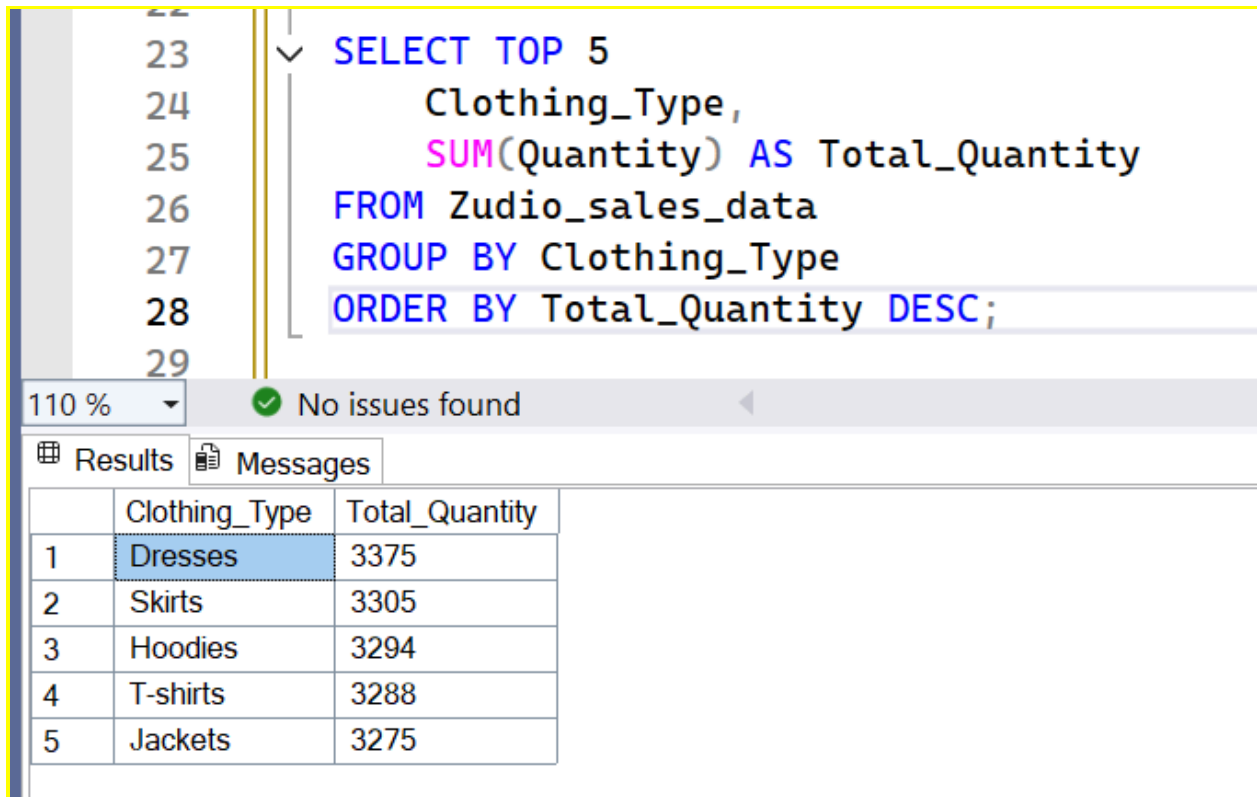
Below the editor, a status bar indicates "110 %", a green checkmark icon, and "No issues found".

At the bottom, there are two tabs: "Results" and "Messages". The "Results" tab is active, displaying a table with the following data:

| | Category | Total_Qty_Sold | Total_Sales |
|---|----------|----------------|-------------|
| 1 | Kids | 12013 | 4707479 |
| 2 | Men | 11727 | 4661546 |
| 3 | Women | 11959 | 4542010 |

3. Top 5 Best-Selling Products

```
SELECT TOP 5
    Clothing_Type,
    SUM(Quantity) AS Total_Quantity
FROM Zudio_sales_data
GROUP BY Clothing_Type
ORDER BY Total_Quantity DESC;
```



The screenshot shows a SQL query editor with a query window and a results pane. The query window displays the following SQL query:

```
SELECT TOP 5
    Clothing_Type,
    SUM(Quantity) AS Total_Quantity
FROM Zudio_sales_data
GROUP BY Clothing_Type
ORDER BY Total_Quantity DESC;
```

Below the query window, there is a status bar indicating "110 %", a green checkmark icon, and the text "No issues found".

The results pane is active, showing a table with the following data:

| | Clothing_Type | Total_Quantity |
|---|---------------|----------------|
| 1 | Dresses | 3375 |
| 2 | Skirts | 3305 |
| 3 | Hoodies | 3294 |
| 4 | T-shirts | 3288 |
| 5 | Jackets | 3275 |

4. Sales by State

```
SELECT
    State,
    SUM(Price) AS Total_Sales
FROM Zudio_sales_data
GROUP BY State
ORDER BY Total_Sales DESC;
```

The screenshot shows a SQL IDE interface. On the left, a vertical list of line numbers from 31 to 37 is visible. To the right of these numbers, the SQL query is displayed and highlighted in blue. Below the query editor, there is a status bar showing '110 %' zoom and a green checkmark icon with the text 'No issues found'. Below the status bar, there are two tabs: 'Results' and 'Messages'. The 'Results' tab is active, displaying a table with 3 columns: 'State', 'Total_Sales', and an implicit index column. The table contains 8 rows of data, with 'Rajasthan' having the highest total sales.

| | State | Total_Sales |
|---|---------------|-------------|
| 1 | Rajasthan | 1803356 |
| 2 | Uttar Pradesh | 1785224 |
| 3 | West Bengal | 1762483 |
| 4 | Delhi | 1758945 |
| 5 | Gujarat | 1716392 |
| 6 | Tamil Nadu | 1705167 |
| 7 | Maharashtra | 1689805 |
| 8 | Karnataka | 1689663 |

5. Contribution by Product Type (Tshirt, Kurti, Jeans, etc.)

```
SELECT
    Clothing_Type,
    SUM(Price) AS Total_Sales
FROM Zudio_sales_data
GROUP BY Clothing_Type
ORDER BY Total_Sales DESC;
```

41 SELECT
42 Clothing_Type,
43 SUM(Price) AS Total_Sales
44 FROM Zudio_sales_data
45 GROUP BY Clothing_Type
46 ORDER BY Total_Sales DESC;
47

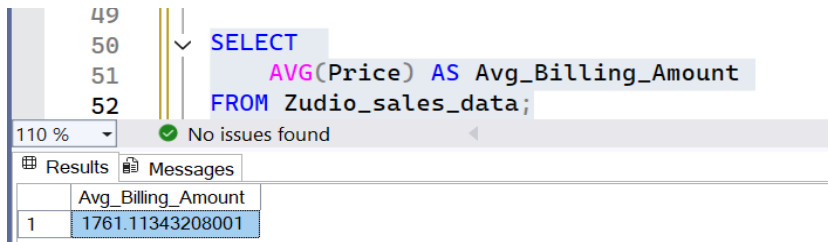
110 % No issues found

Results Messages

| | Clothing_Type | Total_Sales |
|----|---------------|-------------|
| 1 | Dresses | 1320479 |
| 2 | T-shirts | 1301328 |
| 3 | Skirts | 1293344 |
| 4 | Jackets | 1288330 |
| 5 | Pants | 1287756 |
| 6 | Shirts | 1274970 |
| 7 | Sweaters | 1258260 |
| 8 | Tops | 1238021 |
| 9 | Jeans | 1218038 |
| 10 | Shoes | 1215784 |
| 11 | Hoodies | 1214725 |

6. Average Billing Amount Per Customer

```
SELECT
    AVG(Price) AS Avg_Billing_Amount
FROM Zudio_sales_data;
```



The screenshot shows a SQL query editor with the following query:

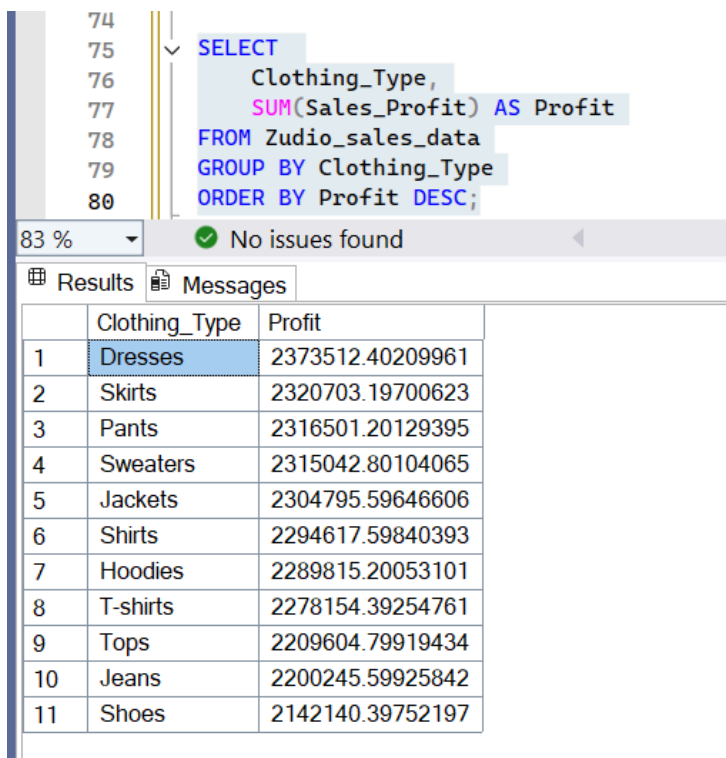
```
SELECT
    AVG(Price) AS Avg_Billing_Amount
FROM Zudio_sales_data;
```

Below the query editor, there is a status bar indicating "No issues found". Below that, there are tabs for "Results" and "Messages". The "Results" tab is active, showing a table with one column, "Avg_Billing_Amount", and one row with the value "1761.11343208001".

| | Avg_Billing_Amount |
|---|--------------------|
| 1 | 1761.11343208001 |

7. Profit Analysis

```
SELECT
    Clothing_Type,
    SUM(Sales_Profit) AS Profit
FROM Zudio_sales_data
GROUP BY Clothing_Type
ORDER BY Profit DESC;
```



The screenshot shows a SQL query editor with the following query:

```
SELECT
    Clothing_Type,
    SUM(Sales_Profit) AS Profit
FROM Zudio_sales_data
GROUP BY Clothing_Type
ORDER BY Profit DESC;
```

Below the query editor, there is a status bar indicating "No issues found". Below that, there are tabs for "Results" and "Messages". The "Results" tab is active, showing a table with two columns, "Clothing_Type" and "Profit", and 11 rows of data.

| | Clothing_Type | Profit |
|----|---------------|------------------|
| 1 | Dresses | 2373512.40209961 |
| 2 | Skirts | 2320703.19700623 |
| 3 | Pants | 2316501.20129395 |
| 4 | Sweaters | 2315042.80104065 |
| 5 | Jackets | 2304795.59646606 |
| 6 | Shirts | 2294617.59840393 |
| 7 | Hoodies | 2289815.20053101 |
| 8 | T-shirts | 2278154.39254761 |
| 9 | Tops | 2209604.79919434 |
| 10 | Jeans | 2200245.59925842 |
| 11 | Shoes | 2142140.39752197 |

8. Stored Type Contribution

```
SELECT
    Store_Type,
    SUM(Price) AS Total_Sales
FROM Zudio_sales_data
GROUP BY Store_Type;
```

The screenshot shows a SQL IDE interface. On the left, a vertical pane lists line numbers 83 through 89. The main editor area displays the following SQL query:

```
SELECT
    Store_Type,
    SUM(Price) AS Total_Sales
FROM Zudio_sales_data
GROUP BY Store_Type;
```

Below the editor, a status bar shows "83 %" zoom and a green checkmark icon with the text "No issues found".

Below the status bar, there are two tabs: "Results" (active) and "Messages". The "Results" tab displays a table with the following data:

| | Store_Type | Total_Sales |
|---|------------|-------------|
| 1 | Rented | 6987234 |
| 2 | Owned | 6923801 |