

TASK-06

Sales trend analysis using SQL Aggrigations

1. Dataset

Limit to 1000 rows

```
select * from orders;
```

| | Order_Id | Order_Date | Ship Mode | Segment | Country | City | State | Postal Code | Region | Category | Sub_Category | Product Id | cost_p |
|---|----------|---------------------|----------------|-------------|---------------|---------------|----------------|-------------|--------|-----------------|--------------|-----------------|--------|
| ▶ | 568 | 2022-01-01 00:00:00 | Standard Class | Corporate | United States | Seattle | Washington | 98105 | West | Office Supplies | Binders | OFF-BI-10001107 | 70 |
| | 8061 | 2022-01-01 00:00:00 | Standard Class | Corporate | United States | Long Beach | New York | 11561 | East | Office Supplies | Paper | OFF-PA-10001970 | 60 |
| | 3442 | 2022-01-01 00:00:00 | Second Class | Consumer | United States | New York City | New York | 10024 | East | Office Supplies | Paper | OFF-PA-10002137 | 100 |
| | 3791 | 2022-01-01 00:00:00 | Standard Class | Home Office | United States | Los Angeles | California | 90004 | West | Office Supplies | Binders | OFF-BI-10001670 | 150 |
| | 2472 | 2022-01-01 00:00:00 | Standard Class | Consumer | United States | Fayetteville | North Carolina | 28314 | South | Furniture | Bookcases | FUR-BO-10002213 | 410 |
| | 9663 | 2022-01-01 00:00:00 | Standard Class | Home Office | United States | Santa Barbara | California | 93101 | West | Technology | Phones | TEC-PH-10001760 | 430 |
| | 1203 | 2022-01-01 00:00:00 | Standard Class | Corporate | United States | Denver | Colorado | 80219 | West | Furniture | Chairs | FUR-CH-10002331 | 770 |
| | 4279 | 2022-01-01 00:00:00 | Standard Class | Corporate | United States | Lakewood | New Jersey | 8701 | East | Furniture | Furnishings | FUR-FU-10000550 | 10 |
| | 8388 | 2022-01-01 00:00:00 | Standard Class | Consumer | United States | Manchester | Connecticut | 6040 | East | Office Supplies | Storage | OFF-ST-10001328 | 60 |
| | 6563 | 2022-01-01 00:00:00 | Standard Class | Consumer | United States | Lawrence | Massachusetts | 1841 | East | Office Supplies | Binders | OFF-BI-10001078 | 20 |
| | 3778 | 2022-01-01 00:00:00 | Standard Class | Home Office | United States | Chula Vista | California | 91911 | West | Furniture | Furnishings | FUR-FU-10002298 | 620 |

2. Extracting months from Order_date

SQL File 6* Sql Practice SQL_Practice* x

Limit to 1000 rows

```
select distinct extract(month from order_date) as Months from orders;
```

| Months |
|--------|
| ▶ 1 |
| 2 |
| 3 |
| 4 |
| 5 |
| 6 |
| 7 |
| 8 |
| 9 |
| 10 |
| 11 |
| 12 |

Result 29 x

3. Extracting distinct month names from the data

SQL File 6* Sql Practice **SQL_Practice*** x

Limit to 1000 rows

```
19 • select distinct date_format(order_date, '%M') as months from orders;
20
21
22
23
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: [↗](#)

| months |
|-----------|
| January |
| February |
| March |
| April |
| May |
| June |
| July |
| August |
| September |
| October |
| November |
| December |

Result 30 x

4. Total number of products ordered in each month for 2 years

SQL File 6* Sql Practice **SQL_Practice*** x

Limit to 1000 rows

```
22
23
24 • select date_format(order_date, '%M') as months, count(order_id)
25 from orders group by months;
26
27
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: [↗](#)

| months | count(order_id) |
|-----------|-----------------|
| January | 858 |
| February | 800 |
| March | 835 |
| April | 848 |
| May | 821 |
| June | 783 |
| July | 905 |
| August | 858 |
| September | 737 |
| October | 861 |
| November | 836 |
| December | 852 |

Result 33 x

5. Total sales in each particular year

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

```
34
35 • select date_format(order_date, '%Y') as years, sum(total_cost) as Total_sales
36     from orders
37     group by years;
38
39
```

Below the editor is the 'Result Grid' section, which includes a 'Filter Rows' input and an 'Export' button. The query results are displayed in a table:

| | years | Total_sales |
|---|-------|-------------|
| ▶ | 2022 | 4958470 |
| | 2023 | 5080930 |

6. Total sales from each category in each year

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

```
45
46
47
48
49 • select category,date_format(order_date, '%Y') as years,sum(total_cost) as total_sales
50     from orders
51     group by category,years
52     order by category;
53
```

Below the editor is the 'Result Grid' section, which includes a 'Filter Rows' input and an 'Export' button. The query results are displayed in a table:

| | category | years | total_sales |
|---|-----------------|-------|-------------|
| ▶ | Furniture | 2022 | 1736760 |
| | Furniture | 2023 | 1641200 |
| | Office Supplies | 2022 | 1502810 |
| | Office Supplies | 2023 | 1603440 |
| | Technology | 2022 | 1718900 |
| | Technology | 2023 | 1836290 |

7. Top 5 most contributing cities in sales in 2nd half of the year 2022

SQL File 6* Sql Practice SQL_Practice* x

Limit to 1000 rows

```
55 • select city, sum(total_cost) as total
56   from orders
57   where order_date < '2022-12-30' and order_date > '2022-06-01'
58   group by city order by total desc limit 5;
59
60
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: | Fetch rows:

| | city | total |
|---|---------------|--------|
| ▶ | New York City | 270650 |
| | Los Angeles | 228500 |
| | Seattle | 221240 |
| | San Francisco | 192870 |
| | Philadelphia | 117840 |

8. Top 5 Sub Categories by Revenue

SQL File 6* Sql Practice SQL_Practice* x

Limit to 1000 rows

```
61
62 • select Sub_Category, sum(quantity) as total_items
63   from orders
64   where order_date > '2022-01-01' and order_date < '2023-01-01'
65   group by sub_category
66   order by total_items desc
67   limit 5;
68
69
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: | Fetch rows:

| | Sub_Category | total_items |
|---|--------------|-------------|
| ▶ | Binders | 3034 |
| | Paper | 2620 |
| | Furnishings | 1850 |
| | Phones | 1613 |
| | Storage | 1573 |

9. Orders Count by Ship Mode

SQL File 6* Sql Practice SQL_Practice* x

Limit to 1000 rows

```
68
69
70 • SELECT `Ship Mode`, COUNT(Order_Id) AS order_count
71 FROM orders
72 GROUP BY `Ship Mode`;
73
74
75
76
```

Result Grid Filter Rows: Export: Wrap Cell Content:

| | Ship Mode | order_count |
|---|----------------|-------------|
| ▶ | Standard Class | 5962 |
| | Second Class | 1945 |
| | Same Day | 543 |
| | First Class | 1538 |
| | Not Available | 4 |
| | N/A | 1 |
| | unknown | 1 |