


Sales Analyse monthly revenue and order volume

```
1 • use ecommerce;
2 • CREATE TABLE orderss (
3     order_id INT PRIMARY KEY,
4     order_date DATE,
5     amount DECIMAL(10, 2),
6     product_id INT
7 );
8
9 • INSERT INTO orderss (order_id, order_date, amount, product_id) VALUES
10 (101, '2023-01-15', 250.00, 1),
11 (102, '2023-01-20', 320.00, 2),
12 (103, '2023-02-05', 180.00, 3),
13 (104, '2023-02-15', 450.00, 1),
14 (105, '2023-03-10', 100.00, 2),
15 (106, '2023-03-22', 220.00, 1),
16 (107, '2023-04-03', 310.00, 4),
17 (108, '2023-04-25', 410.00, 3),
18 (109, '2023-05-05', 90.00, 2),
19 (110, '2023-05-14', 270.00, 4),
20 (111, '2023-06-01', 330.00, 1),
21 (112, '2023-06-10', 190.00, 3),
22 (113, '2023-07-11', 360.00, 2),
23 (114, '2023-07-18', 140.00, 3),
24 (115, '2023-08-01', 280.00, 4);
```

Result Grid				
		Filter Rows:	Edit	
	order_id	order_date	amount	product_id
▶	101	2023-01-15	250.00	1
	102	2023-01-20	320.00	2
	103	2023-02-05	180.00	3
	104	2023-02-15	450.00	1
	105	2023-03-10	100.00	2
	106	2023-03-22	220.00	1
	107	2023-04-03	310.00	4
	108	2023-04-25	410.00	3
	109	2023-05-05	90.00	2
	110	2023-05-14	270.00	4
	111	2023-06-01	330.00	1
	112	2023-06-10	190.00	3
	113	2023-07-11	360.00	2
	114	2023-07-18	140.00	3
	115	2023-08-01	280.00	4
*	NULL	NULL	NULL	NULL



Extract (month from Order Date)

```
28 -- EXTRACT(MONTH FROM order_date)
29 • SELECT
30     EXTRACT(MONTH FROM order_date) AS month_number,
31     COUNT(*) AS total_orders FROM orderss
32 GROUP BY EXTRACT(MONTH FROM order_date);
```

Result Grid		
Filter Rows: <input type="text"/>		
Export:  Wrap Cell		
	month_number	total_orders
▶	1	2
	2	2
	3	2
	4	2
	5	2
	6	2
	7	2
	8	1





Group by Year and Month

```
34 -- Group by Year and Month
35 • SELECT
36     YEAR(order_date) AS order_year,
37     MONTH(order_date) AS order_month,
38     COUNT(*) AS orders_count FROM orderss
39 GROUP BY YEAR(order_date), MONTH(order_date);
40
```

Result Grid			
Filter Rows: <input type="text"/>			
Export:  Wrap Cell Content: 			
	order_year	order_month	orders_count
▶	2023	1	2
	2023	2	2
	2023	3	2
	2023	4	2
	2023	5	2
	2023	6	2
	2023	7	2
	2023	8	1

Use SUM() for Total Revenue





```
41  -- Use SUM() for Total Revenue
42 •  SELECT
43      YEAR(order_date) AS year,
44      MONTH(order_date) AS month,
45      SUM(amount) AS total_revenue
46  FROM orderss
47  GROUP BY YEAR(order_date), MONTH(order_date);
```

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

	year	month	total_revenue
▶	2023	1	570.00
	2023	2	630.00
	2023	3	320.00
	2023	4	720.00
	2023	5	360.00
	2023	6	520.00
	2023	7	500.00
	2023	8	280.00

Use COUNT(DISTINCT order_id) for Volume

```
49  -- Use COUNT(DISTINCT order_id) for Volume
50 •  SELECT
51      EXTRACT(YEAR_MONTH FROM order_date) AS years_month,
52      COUNT(DISTINCT order_id) AS order_volume
53  FROM orderss
54  GROUP BY years_month;
```

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

	years_month	order_volume
▶	202301	2
	202302	2
	202303	2
	202304	2
	202305	2
	202306	2
	202307	2
	202308	1

Use ORDER BY for Sorting

```
56  -- Use ORDER BY for Sorting
57 •  SELECT
58      product_id,
59      SUM(amount) AS revenue
60  FROM orderss
61  GROUP BY product_id
62  ORDER BY revenue DESC;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Co

	product_id	revenue
▶	1	1250.00
	3	920.00
	2	870.00
	4	860.00

Limit Results for Specific Time Periods




```
64  -- Limit Results for Specific Time Periods
65 •  SELECT * FROM orderss
66  WHERE order_date BETWEEN '2023-01-01' AND '2023-06-30';
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

	order_id	order_date	amount	product_id
▶	101	2023-01-15	250.00	1
	102	2023-01-20	320.00	2
	103	2023-02-05	180.00	3
	104	2023-02-15	450.00	1
	105	2023-03-10	100.00	2
	106	2023-03-22	220.00	1
	107	2023-04-03	310.00	4
	108	2023-04-25	410.00	3
	109	2023-05-05	90.00	2

Rolling 3-Month Revenue




```
68  -- Rolling 3-Month Revenue
69  • SELECT
70      DATE_FORMAT(order_date, '%Y-%m') AS years_month,
71      SUM(amount) AS monthly_revenue FROM orderss
72  WHERE order_date >= DATE_SUB('2023-08-01', INTERVAL 3 MONTH)
73  GROUP BY years_month
74  ORDER BY years_month;
```

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

	years_month	monthly_revenue
▶	2023-05	360.00
	2023-06	520.00
	2023-07	500.00
	2023-08	280.00

Grouping online sales by year and month to analyse revenue and order volume

```
76  -- Grouping online sales by year and month to analyze revenue and order volume
77  • SELECT
78      YEAR(order_date) AS order_year, MONTH(order_date) AS order_month,
79      COUNT(DISTINCT order_id) AS total_orders,
80      SUM(amount) AS total_revenue, AVG(amount) AS avg_order_value,
81      MIN(amount) AS min_order_value, MAX(amount) AS max_order_value
82  FROM orderss
83  WHERE order_date BETWEEN '2023-01-01' AND '2023-12-31'
84  GROUP BY YEAR(order_date), MONTH(order_date)
85  ORDER BY YEAR(order_date), MONTH(order_date);
```

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

	order_year	order_month	total_orders	total_revenue	avg_order_value	min_order_value	max_order_value
▶	2023	1	2	570.00	285.000000	250.00	320.00
	2023	2	2	630.00	315.000000	180.00	450.00
	2023	3	2	320.00	160.000000	100.00	220.00
	2023	4	2	720.00	360.000000	310.00	410.00
	2023	5	2	360.00	180.000000	90.00	270.00
	2023	6	2	520.00	260.000000	190.00	330.00
	2023	7	2	500.00	250.000000	140.00	360.00