

OUTPUT SCREENSHOTS

1. Extract month and year from order_date

The screenshot shows a SQL IDE window with a query editor and a result grid. The query editor contains the following SQL code:

```
1 • USE order_analysis;  
2 • CREATE TABLE orders (  
3     order_date DATE,  
4     product_id INT,  
5     city_id INT,  
6     orders INT  
7 );  
8  
9 -- Extract month and year from order_date  
10 • SELECT  
11     EXTRACT(YEAR FROM order_date) AS order_year,  
12     EXTRACT(MONTH FROM order_date) AS order_month  
13 FROM orders;
```

The result grid displays the following data:

order_year	order_month
2019	12
2018	8
2018	10
2019	8
2019	1
2018	8
2018	11
2019	3
2019	6

2. Monthly Revenue

The screenshot shows a SQL IDE window with a query editor and a result grid. The query editor contains the following SQL code:

```
15 -- Monthly Revenue  
16 • SELECT  
17     EXTRACT(YEAR FROM order_date) AS order_year,  
18     EXTRACT(MONTH FROM order_date) AS order_month,  
19     SUM(orders) AS total_revenue  
20 FROM orders  
21 GROUP BY order_year, order_month  
22 ORDER BY order_year, order_month;  
23
```

The result grid displays the following data:

order_year	order_month
2019	12
2018	8
2018	10
2019	8
2019	1
2018	8
2018	11
2019	3
2019	6
2018	8
2019	10
2019	8
2019	3
2018	8
2019	6

3. Group by year and month & calculate total volume

SQL File 3* x

Limit to 1000 rows

```

24  -- Group by year and month & calculate total volume
25  •  SELECT
26      EXTRACT(YEAR FROM order_date) AS order_year,
27      EXTRACT(MONTH FROM order_date) AS order_month,
28      SUM(orders) AS total_volume
29  FROM orders
30  GROUP BY order_year, order_month
31  ORDER BY order_year, order_month;
32

```

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

	order_year	order_month	total_volume
▶	2018	7	780
	2018	8	1839
	2018	9	817
	2018	10	366
	2018	11	1057
	2018	12	1274
	2019	1	596
	2019	2	525
	2019	3	698
	2019	4	1354
	2019	5	1802
	2019	6	606
	2019	7	1376
	2019	8	1044
	2019	9	17

4. Limit results to a specific time period (e.g., 2019 only)

Limit to 1000 rows

```

36  -- Limit results to a specific time period (e.g., 2019 only)
37  •  SELECT
38      EXTRACT(YEAR FROM order_date) AS order_year,
39      EXTRACT(MONTH FROM order_date) AS order_month,
40      SUM(orders) AS total_volume
41  FROM orders
42  WHERE YEAR(order_date) = 2019
43  GROUP BY order_year, order_month
44  ORDER BY order_month;
45

```

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

	order_year	order_month	total_volume
▶	2019	1	596
	2019	2	525
	2019	3	698
	2019	4	1354
	2019	5	1802
	2019	6	606
	2019	7	1376
	2019	8	1044
	2019	9	17
	2019	10	1141
	2019	11	798
	2019	12	1407

Result 13 x

5. Count number of distinct product orders per month

SQL File 3*

```
47 -- Count number of distinct product orders per month
48 • SELECT
49     EXTRACT(YEAR FROM order_date) AS order_year,
50     EXTRACT(MONTH FROM order_date) AS order_month,
51     COUNT(DISTINCT product_id) AS unique_products
52 FROM orders
53 GROUP BY order_year, order_month
54 ORDER BY order_year, order_month;
55
56
```

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

	order_year	order_month	unique_products
▶	2018	7	27
	2018	8	58
	2018	9	63
	2018	10	61
	2018	11	67
	2018	12	75
	2019	1	57
	2019	2	42
	2019	3	48
	2019	4	55
	2019	5	42
	2019	6	63
	2019	7	79

Result 14 x