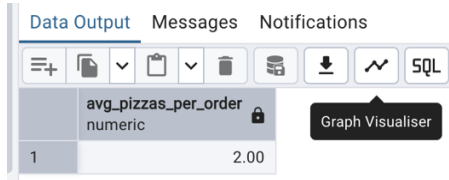


The image shows a software interface with three tabs: 'Data Output', 'Messages', and 'Notifications'. The 'Data Output' tab is selected and highlighted with a blue underline. Below the tabs is a toolbar containing several icons: a plus sign, a document icon, a dropdown arrow, a clipboard icon, a trash can icon, a database cylinder icon, a download icon, a line graph icon, and a 'SQL' button. Below the toolbar is a table. The table has two columns. The first column is labeled 'total_orders' and the second column is labeled 'bigint'. The first row of data shows the value '1' under 'total_orders' and '21350' under 'bigint'. The 'total_orders' header has a lock icon next to it.

-- 5. Average Pizzas Per Order

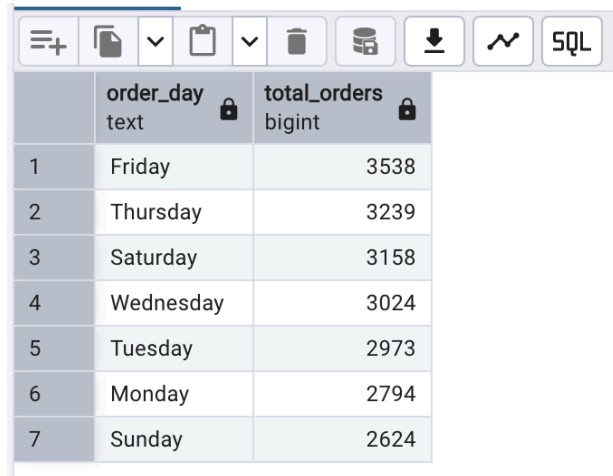
```
SELECT ROUND(SUM(quantity) / COUNT(DISTINCT order_id), 2) AS avg_pizzas_per_order  
FROM pizza_sales;
```



	avg_pizzas_per_order numeric
1	2.00

B. Daily Trend for Total Orders

```
SELECT TO_CHAR (order_date, 'Day') AS order_day, COUNT (DISTINCT order_id) AS  
total_orders  
FROM pizza_sales  
GROUP BY order_day  
ORDER BY total_orders DESC;
```



	order_day text	total_orders bigint
1	Friday	3538
2	Thursday	3239
3	Saturday	3158
4	Wednesday	3024
5	Tuesday	2973
6	Monday	2794
7	Sunday	2624

C. Hourly Trend for Orders

	order_hour numeric	totalOrders bigint
1	9	1
2	10	8
3	11	1231
4	12	2520
5	13	2455
6	14	1472
7	15	1468
8	16	1920
9	17	2336
10	18	2399
11	19	2009
12	20	1642
13	21	1198
14	22	663
15	23	28

D. % of Sales by Pizza Category

```
SELECT pizza_category, ROUND(SUM(total_price)::numeric, 2) AS total_revenue,  
ROUND((SUM(total_price) * 100) / (SELECT SUM(total_price) FROM pizza_sales), 2) AS  
percentage_sales  
FROM pizza_sales  
GROUP BY pizza_category  
ORDER BY percentage_sales DESC;
```

	pizza_category character varying (20)	total_revenue numeric	percentage_sales numeric
1	Classic	220053.10	26.91
2	Supreme	208197.00	25.46
3	Chicken	195919.50	23.96
4	Veggie	193690.45	23.68

E. % of Sales by Pizza Size

```
SELECT pizza_size, ROUND(SUM(total_price)::numeric, 2) AS total_revenue,  
ROUND((SUM(total_price) * 100) / (SELECT SUM(total_price) FROM pizza_sales), 2) AS  
percentage_sales  
  
FROM pizza_sales  
  
GROUP BY pizza_size  
  
ORDER BY pizza_size;
```

	pizza_size character varying (5)	total_revenue numeric	percentage_sales numeric
1	L	375318.70	45.89
2	M	249382.25	30.49
3	S	178076.50	21.77
4	XL	14076.00	1.72
5	XXL	1006.60	0.12

F. Total Pizzas Sold by Pizza Category (for a specific month)

```
SELECT pizza_category, SUM(quantity) AS total_quantity_sold  
  
FROM pizza_sales  
  
WHERE EXTRACT(MONTH FROM order_date) = 2  
  
GROUP BY pizza_category  
  
ORDER BY total_quantity_sold DESC;
```

	pizza_category character varying (20) 🔒	total_quantity_sold bigint 🔒
1	Classic	1178
2	Supreme	964
3	Veggie	944
4	Chicken	875

G. Top 5 Best Sellers by Total Pizzas Sold

```
SELECT pizza_name, SUM(quantity) AS total_pizza_sold
FROM pizza_sales
GROUP BY pizza_name
ORDER BY total_pizza_sold DESC
LIMIT 5;
```

	pizza_name character varying (100) 🔒	total_pizza_sold bigint 🔒
1	The Classic Deluxe Pizza	2453
2	The Barbecue Chicken Pizza	2432
3	The Hawaiian Pizza	2422
4	The Pepperoni Pizza	2418
5	The Thai Chicken Pizza	2371

H. Bottom 5 Best Sellers by Total Pizzas Sold

```
SELECT pizza_name, SUM(quantity) AS total_pizza_sold
FROM pizza_sales
GROUP BY pizza_name
```

ORDER BY total_pizza_sold ASC

LIMIT 5;

	pizza_name character varying (100)	total_pizza_sold bigint
1	The Brie Carre Pizza	490
2	The Mediterranean Pizza	934
3	The Calabrese Pizza	937
4	The Spinach Supreme Pizza	950
5	The Soppressata Pizza	961

Filtering by Month, Quarter, or Week

-- Example: Filtering by January (Month = 1)

```
SELECT TO_CHAR(order_date, 'Day') AS order_day, COUNT(DISTINCT order_id) AS  
total_orders
```

```
FROM pizza_sales
```

```
WHERE EXTRACT(MONTH FROM order_date) = 1
```

```
GROUP BY order_day
```

```
ORDER BY total_orders DESC;
```

	order_day text	total_orders bigint
1	Friday	330
2	Thursday	329
3	Saturday	303
4	Tuesday	242
5	Wednesday	222
6	Monday	220
7	Sunday	199

-- Example: Filtering by Quarter 1

```
SELECT TO_CHAR(order_date, 'Day') AS order_day, COUNT(DISTINCT order_id) AS  
total_orders
```

```
FROM pizza_sales
```

```
WHERE EXTRACT(QUARTER FROM order_date) = 1
```

```
GROUP BY order_day
```

```
ORDER BY total_orders DESC;
```

	order_day text	total_orders bigint
1	Friday	885
2	Thursday	796
3	Tuesday	787
4	Saturday	766
5	Monday	743
6	Wednesday	711
7	Sunday	682