

MS-SQL Case Study 3 (Burger Bash)

-Sarthak Niranjan Kulkarni (Maverick)

- sarthakkul2311@gmail.com

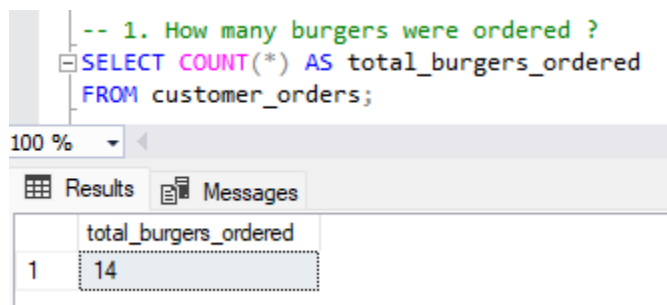
- (+91) 93256 02791

Day 4 – 7/11/2024 (Thursday)

1. How many burgers were ordered?

→ SELECT COUNT(*) AS total_burgers_ordered

FROM customer_orders;



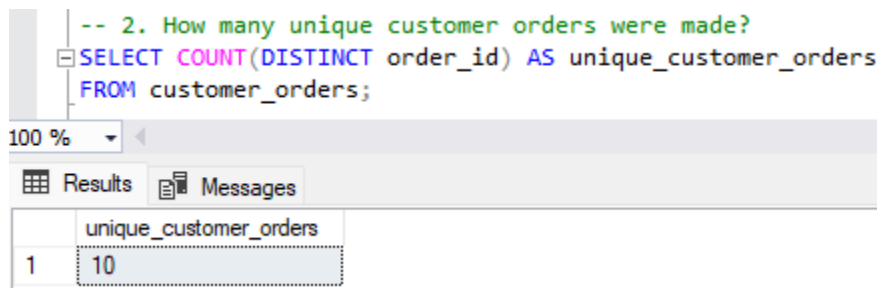
The screenshot shows a SQL query editor with the following text: `-- 1. How many burgers were ordered ?`
`SELECT COUNT(*) AS total_burgers_ordered`
`FROM customer_orders;`
Below the query, there is a 'Results' tab showing a single row with the column 'total_burgers_ordered' and the value '14'.

	total_burgers_ordered
1	14

2. How many unique customer orders were made?

→ SELECT COUNT(DISTINCT order_id) AS unique_customer_orders

FROM customer_orders;



The screenshot shows a SQL query editor with the following text: `-- 2. How many unique customer orders were made?`
`SELECT COUNT(DISTINCT order_id) AS unique_customer_orders`
`FROM customer_orders;`
Below the query, there is a 'Results' tab showing a single row with the column 'unique_customer_orders' and the value '10'.

	unique_customer_orders
1	10

3. How many successful orders were delivered by each runner?

→ SELECT runner_id, COUNT(*) AS successful_orders

FROM runner_orders

WHERE cancellation IS NULL

GROUP BY runner_id;

```
-- 3. How many successful orders were delivered by each runner?  
SELECT runner_id, COUNT(*) AS successful_orders  
FROM runner_orders  
WHERE cancellation IS NULL  
GROUP BY runner_id;
```

	runner_id	successful_orders
1	1	4
2	2	3
3	3	1

4. How many of each type of burger was delivered?

→ SELECT b.burger_name, COUNT(co.order_id) AS burger_count

FROM customer_orders co

JOIN burger_names b ON co.burger_id = b.burger_id

GROUP BY b.burger_name;

```
-- 4. How many of each type of burger was delivered?  
SELECT b.burger_name, COUNT(co.order_id) AS burger_count  
FROM customer_orders co  
JOIN burger_names b ON co.burger_id = b.burger_id  
GROUP BY b.burger_name;
```

	burger_name	burger_count
1	Meatlovers	10
2	Vegetarian	4

5. How many Vegetarian and Meatlovers were ordered by each customer?

→ SELECT co.customer_id, b.burger_name, COUNT(co.order_id) AS burger_count

FROM customer_orders co

JOIN burger_names b ON co.burger_id = b.burger_id

GROUP BY co.customer_id, b.burger_name

ORDER BY co.customer_id, b.burger_name;

```
-- 5. How many Vegetarian and Meatlovers were ordered by each customer?  
SELECT co.customer_id, b.burger_name, COUNT(co.order_id) AS burger_count  
FROM customer_orders co  
JOIN burger_names b ON co.burger_id = b.burger_id  
GROUP BY co.customer_id, b.burger_name  
ORDER BY co.customer_id, b.burger_name;
```

	customer_id	burger_name	burger_count
1	101	Meatlovers	2
2	101	Vegetarian	1
3	102	Meatlovers	2
4	102	Vegetarian	1
5	103	Meatlovers	3
6	103	Vegetarian	1
7	104	Meatlovers	3
8	105	Vegetarian	1

6. What was the maximum number of burgers delivered in a single order?

→ SELECT MAX(order_count) AS max_burgers_in_single_order

FROM (SELECT order_id, COUNT(burger_id) AS order_count

FROM customer_orders

GROUP BY order_id) AS burger_counts;

```
-- 6. What was the maximum number of burgers delivered in a single order?  
SELECT MAX(order_count) AS max_burgers_in_single_order  
FROM (SELECT order_id, COUNT(burger_id) AS order_count  
FROM customer_orders  
GROUP BY order_id) AS burger_counts;
```

	max_burgers_in_single_order
1	3

7. What was the total volume of burgers ordered for each hour of the day?

→

```
SELECT DATEPART(hour, order_time) AS order_hour, COUNT(order_id) AS  
total_burgers_ordered
```

```
FROM customer_orders
```

```
GROUP BY DATEPART(hour, order_time)
```

```
ORDER BY order_hour;
```

```
-- 7. What was the total volume of burgers ordered for each hour of the day?  
SELECT DATEPART(hour, order_time) AS order_hour, COUNT(order_id) AS total_burgers_ordered  
FROM customer_orders  
GROUP BY DATEPART(hour, order_time)  
ORDER BY order_hour;
```

100 %

Results Messages

	order_hour	total_burgers_ordered
1	11	1
2	13	3
3	18	3
4	19	1
5	21	3
6	23	3
