

# PIZZA RUNNERS



# INTRODUCTION



Did you know that over 115 million kilograms of pizza is consumed daily worldwide??? (Well according to Wikipedia anyway...)

Danny was scrolling through his Instagram feed when something really caught his eye - “80s Retro Styling and Pizza Is The Future!”

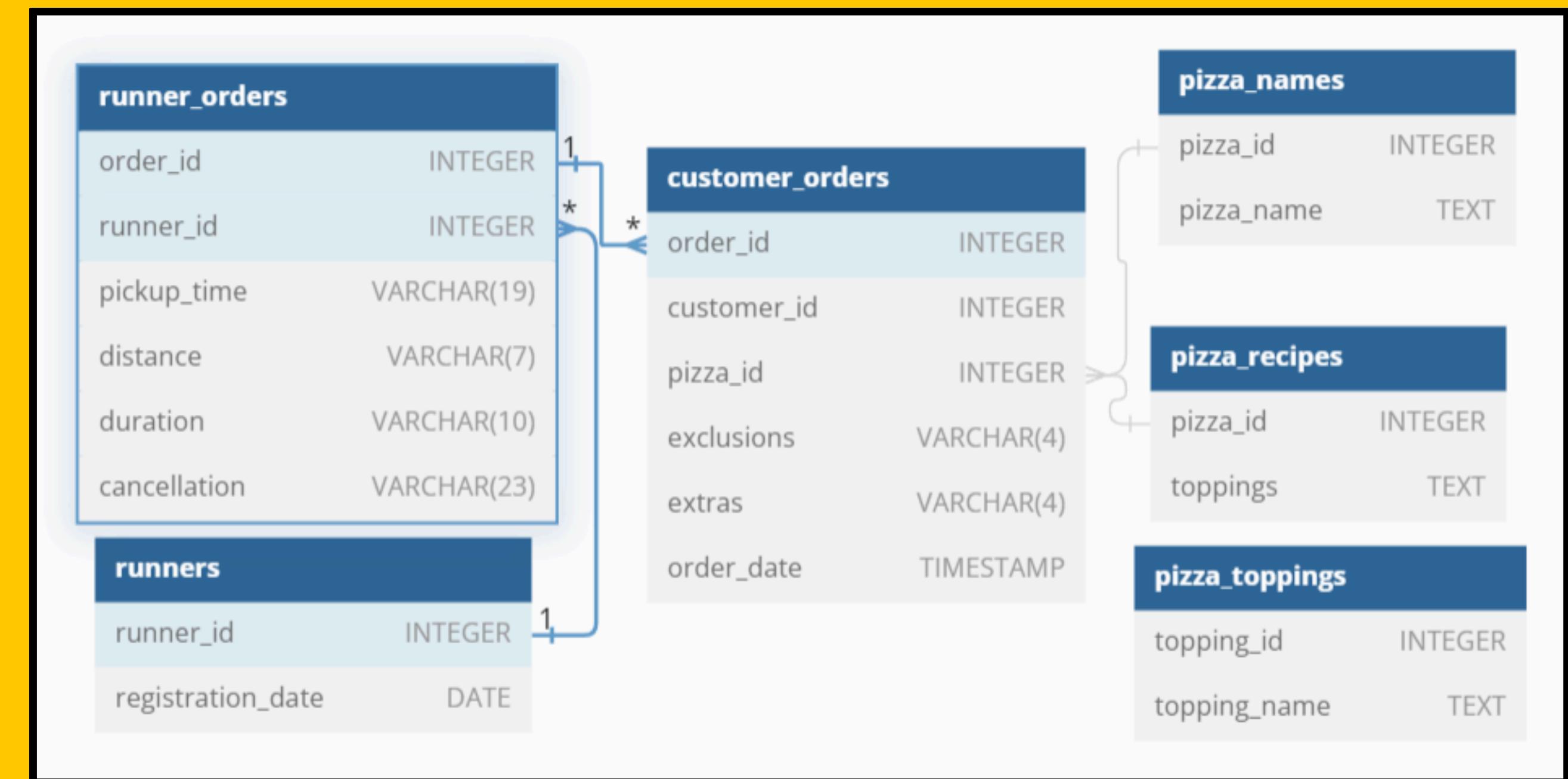


Danny was sold on the idea, but he knew that pizza alone was not going to help him get seed funding to expand his new Pizza Empire – so he had one more genius idea to combine with it – he was going to Uberize it – and so Pizza Runner was launched!

Danny started by recruiting “runners” to deliver fresh pizza from Pizza Runner Headquarters (otherwise known as Danny’s house) and also maxed out his credit card to pay freelance developers to build a mobile app to accept orders from customers.



# ENTITY RELATIONSHIP DIAGRAM





1. How many pizzas were ordered?

```
SELECT COUNT(*) AS Pizzas_Ordered  
FROM customer_orders;
```

	Pizzas_Ordered
▶	14



2. How many unique customer orders were made?

```
SELECT COUNT(DISTINCT order_id) AS Order_Count  
FROM customer_orders;
```

	Order_Count
▶	10



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

```
SELECT runner_id,  
       COUNT(DISTINCT order_id) AS Order_Delivered  
FROM runner_orders ro  
WHERE pickup_time <> 'null'  
GROUP BY runner_id;
```

	runner_id	Order_Delivered
▶	1	4
	2	3
	3	1



## 4. How many of each type of pizza was delivered?

```
SELECT pn.pizza_name AS Pizza_Type,  
       COUNT(co.order_id) AS Delivered_Count  
FROM customer_orders co  
JOIN runner_orders ro ON co.order_id = ro.order_id  
JOIN pizza_names pn ON pn.pizza_id = co.pizza_id  
WHERE ro.pickup_time <> 'null'  
GROUP BY pn.pizza_name;
```

	Pizza_Type	Delivered_Count
▶	Meatlovers	9
	Vegetarian	3



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

```
SELECT customer_id,  
       pn.pizza_name,  
       COUNT(co.pizza_id) AS pizzas_ordered  
FROM customer_orders co  
JOIN pizza_names pn ON pn.pizza_id = co.pizza_id  
GROUP BY customer_id, pizza_name  
ORDER BY customer_id;
```

	customer_id	pizza_name	pizzas_ordered
	101	Vegetarian	1
	102	Meatlovers	2
	102	Vegetarian	1
	103	Meatlovers	3
	103	Vegetarian	1
	104	Meatlovers	3
	105	Vegetarian	1



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

```
SELECT co.order_id,
       COUNT(co.order_id) AS Pizzas_Ordered
  FROM customer_orders co
 JOIN runner_orders ro ON co.order_id = ro.order_id
 WHERE ro.pickup_time <> 'null'
 GROUP BY co.order_id
 ORDER BY COUNT(co.order_id) DESC
 LIMIT 1;
```

	order_id	Pizzas_Ordered
▶	4	3



# 7. For each customer, how many delivered pizzas had at least 1 change and how many had no changes?

```
SELECT
    customer_id,
    SUM(
        CASE
            WHEN (exclusions IS NOT NULL AND exclusions <> 'null' AND LENGTH(exclusions) > 0)
                OR (extras IS NOT NULL AND extras <> 'null' AND LENGTH(extras) > 0)
            THEN 1 ELSE 0 END
    ) AS changes,
    SUM(
        CASE
            WHEN (exclusions IS NOT NULL AND exclusions <> 'null' AND LENGTH(exclusions) > 0)
                OR (extras IS NOT NULL AND extras <> 'null' AND LENGTH(extras) > 0)
            THEN 0 ELSE 1 END
    ) AS no_changes
FROM
    customer_orders AS co
    INNER JOIN runner_orders AS ro ON ro.order_id = co.order_id
WHERE ro.pickup_time <> 'null'
GROUP BY customer_id;
```

	customer_id	changes	no_changes
	101	0	2
	102	0	3
	103	3	0
	104	2	1
	105	1	0



## 8. How many pizzas were delivered that had both exclusions and extras?

```
SELECT  
    COUNT(co.order_id) AS Pizzas_With_Exclusion_AND_Extras  
FROM  
    customer_orders co  
    JOIN runner_orders ro ON co.order_id = ro.order_id  
WHERE  
    pickup_time <> 'null'  
    AND (exclusions <> 'null' AND LENGTH(exclusions) > 0)  
    AND (extras <> 'null' AND LENGTH(extras) > 0);
```

	Pizzas_With_Exclusion_AND_Extras
▶	1



9. What was the total volume of pizzas ordered for each hour of the day?

```
SELECT HOUR(order_time) AS Hour,  
       COUNT(order_id) AS Orders  
  FROM customer_orders  
 GROUP BY HOUR(order_time)  
 ORDER BY Hour;
```

	Hour	Orders
▶	11	1
	13	3
	18	3
	19	1
	21	3
	23	3



10. What was the volume of orders for each day of the week?

```
SELECT DAYOFWEEK(order_time) AS Day_Of_Week,  
       COUNT(order_id) AS Orders  
  FROM customer_orders  
 GROUP BY DAYOFWEEK(order_time);
```

	Day_Of_Week	Orders
4	5	5
5	3	3
7	5	5
6	1	1



11. How many runners signed up for each 1 week period? (i.e. week starts 2021-01-01)

```
SELECT DATE_ADD('2021-01-01', INTERVAL FLOOR(DATEDIFF(registration_date, '2021-01-01') / 7) * 7 DAY)
      AS week_start,
      COUNT(runner_id) AS COUNT
FROM Runners
GROUP BY week_start;
```

	week_start	COUNT
▶	2021-01-01	2
	2021-01-08	1
	2021-01-15	1



```
SELECT runner_id,  
       AVG(TIMESTAMPDIFF(MINUTE, co.order_time, ro.pickup_time)) AS avg_pickup_time_minutes  
FROM runner_orders ro  
JOIN customer_orders co ON ro.order_id = co.order_id  
WHERE ro.pickup_time IS NOT NULL  
GROUP BY ro.runner_id  
ORDER BY ro.runner_id;
```

	runner_id	avg_pickup_time_minutes
▶	1	15.3333
	2	23.4000
	3	10.0000



# 13. Is there any relationship between the number of pizzas and how long the order takes to prepare?

```
WITH CTE AS (
    SELECT co.order_id,
           COUNT(pizza_id) AS No_of_pizzas,
           MAX(TIMESTAMPDIFF(MINUTE, co.order_time, ro.pickup_time)) AS prep_time
      FROM runner_orders ro
     JOIN customer_orders co ON ro.order_id = co.order_id
     WHERE ro.pickup_time IS NOT NULL
   GROUP BY co.order_id
)
SELECT No_of_pizzas, AVG(prep_time) AS Avg_Prep_Time
  FROM CTE
 GROUP BY No_of_pizzas;
```

	No_of_pizzas	Avg_Prep_Time
1	12.0000	
2	18.0000	
3	29.0000	



## 14. What was the average distance travelled for each customer?

```
SELECT customer_id,  
       ROUND(AVG(distance), 2) AS Avg_Distance  
FROM runner_orders ro  
JOIN customer_orders co ON ro.order_id = co.order_id  
WHERE distance IS NOT NULL  
GROUP BY customer_id;
```

	customer_id	Avg_Distance
▶	101	13.33
	102	16.73
	103	17.55
	104	10
	105	25



15. What was the difference between the longest and shortest delivery times for all orders?

**SELECT**

```
MAX(REGEXP_REPLACE(duration, '[^0-9]', '')) -  
MIN(REGEXP_REPLACE(duration, '[^0-9]', '')) AS delivery_time_difference  
FROM runner_orders  
WHERE duration IS NOT NULL  
AND duration <> 'null';
```

	delivery_time_difference
▶	30



# 16. What was the average speed for each runner for each delivery and do you notice any trend for these values?

```
SELECT ro.runner_id,  
       co.order_id,  
       ROUND((SUM(CAST(REGEXP_REPLACE(distance, '[^0-9.]', '') AS DECIMAL))/(SUM(CAST(REGEXP_REPLACE(duration, '[^0-9]', '')  
                                         AS DECIMAL))/60)),2) AS ^Avg_Speed(kmph)  
  
FROM runner_orders ro  
JOIN customer_orders co ON ro.order_id = co.order_id  
WHERE pickup_time <> 'null'  
GROUP BY ro.runner_id, co.order_id;
```

	runner_id	order_id	Avg_Speed(kmph)
▶	1	1	37.50
	1	2	44.44
	1	3	39.00
	2	4	34.50
	3	5	40.00
	2	7	60.00
	2	8	92.00
	1	10	60.00



# 17. What is the successful delivery percentage for each runner?

```
WITH CTE AS (
    SELECT
        runner_id,
        COUNT(CASE WHEN pickup_time <> 'null' THEN 1 END) AS successful_deliveries,
        COUNT(CASE WHEN pickup_time = 'null' THEN 1 END) AS unsuccessful_deliveries
    FROM runner_orders
    GROUP BY runner_id
)
SELECT
    *,
    (successful_deliveries / (successful_deliveries + unsuccessful_deliveries)) * 100
    AS Succeful_Delivery_Percentage
FROM CTE;
```

	runner_id	successful_deliveries	unsuccessful_deliveries	Succeful_Delivery_Percentage
▶	1	4	0	100.0000
	2	3	1	75.0000
	3	1	1	50.0000