

8 Week SQL Challenge

Case Study #2

PIZZA RUNNER





Problem Statement



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.



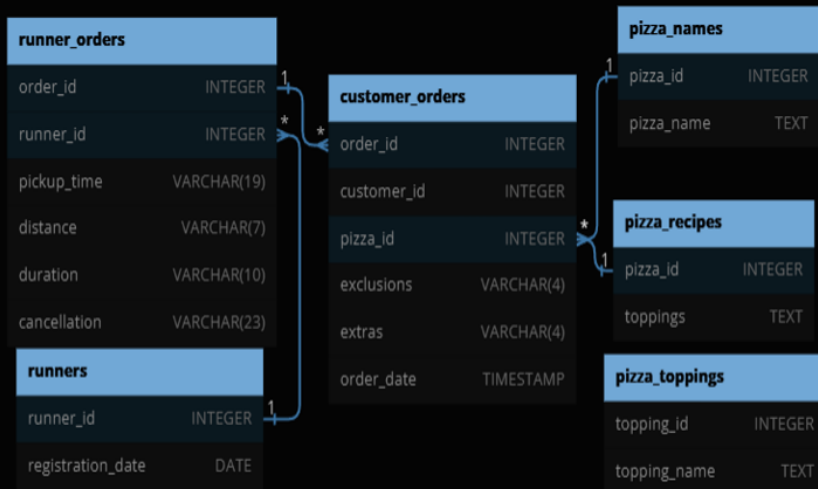
He has prepared for us an entity relationship diagram of his database design but requires further assistance to clean his data and apply some basic calculations so he can better direct his runners and optimize Pizza Runner’s operations.



Tables & ER Diagram



ER Diagram



Powered by dbdiagram.io

Tables



runners



customer_orders



pizza_names



pizza_recipes



pizza_toppings



runner_orders

Data Cleaning

Table: Cleaning runner_orders

👉 Update columns cancellation, pickup_time, distance and duration values which contain "" and "null" to NULL values

👉 Extract numbers from alphanumeric values in columns distance and duration.





👉 Alter the data type for pickup_time to datetime format

👉 Alter the data type for distance and duration to float

Final Table

order_id integer	runner_id integer	pickup_time timestamp without time zone	distance double precision	duration double precision	cancellation character varying
4	2	2020-01-04 13:53:03	23.4	40	[null]
5	3	2020-01-08 21:10:57	10	15	[null]
6	3	[null]	[null]	[null]	Restaurant Cancellation
9	2	[null]	[null]	[null]	Customer Cancellation
3	1	2020-01-03 00:12:37	13.4	20	[null]
1	1	2020-01-01 18:15:34	20	32	[null]
2	1	2020-01-01 19:10:54	20	27	[null]
7	2	2020-01-08 21:30:45	25	25	[null]
8	2	2020-01-10 00:15:02	23.4	15	[null]
10	1	2020-01-11 18:50:20	10	10	[null]

Table: Cleaning pizza_recipes

-  Create a temporary table.
-  Insert data Unnest toppings coulumn into single valued column.
-  Alter data types
-  Make this temporary table as original table by truncating existing values and inserting new values

Final Table


pizza_id integer 	toppings text 
1	1
1	2
1	3
1	4
1	5
1	6
1	8

Table: Cleaning customer_orders

👉 Alter data type of order_time to datetime format.

👉 Create a temporary table to store unnested values of multivalue columns extras and exclusions.

👉 Alter data type of columns present in customer_orders temporary table.

Final Tables

- customer_orders_temp table (temporary)

order_id integer	customer_id integer	pizza_id integer	exclusions integer	extras integer	order_time timestamp without time zone	sno integer
9	103	1	4	1	2020-01-10 11:22:59	1
9	103	1	4	5	2020-01-10 11:22:59	1
10	104	1	2	1	2020-01-11 18:34:49	2
10	104	1	2	4	2020-01-11 18:34:49	2
10	104	1	6	1	2020-01-11 18:34:49	2
10	104	1	6	4	2020-01-11 18:34:49	2
3	102	2	[null]	[null]	2020-01-02 23:51:23	3

Interesting Queries 🤔 & Solutions 😊

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

👉 Query

```
select EXTRACT(hour from order_time) as "Hour", count(order_id) as tot_vol  
from customer_orders  
group by 1  
order by 2 desc;
```

👉 Explanation

- Table used: **customer_orders**
- Extract **hours** from **order_time** using **extract**
- **Count** and **group by** over **hours** using **group by**

👉 Result

Hour numeric	tot_vol bigint
18	3
21	3
23	3
13	3
19	1
11	1

🍕 What is the successful delivery percentage for each runner?

👉 Query

```
with cte as
(
  select runner_id, sum(case when CANCELLATION is NULL then 1 else 0 end) as success,
  count(*) as tot
  from runner_orders
  group by 1
)

select runner_id, (success*100)/(tot) as success_percent
from cte
order by 2 desc;
```

👉 Explanation

- Table used: **runner_orders**
- CTE cte: Gather **count of successful orders** for each runner_id using **case, sum, group by**
- Calculate **success percentage** by using **cte result**.

👉 Result

runner_id integer	success_percent bigint
1	100
2	75
3	50

🍕 What was the most commonly added extra?

👉 Query

```
select pt.topping_name as common_extras, count(distinct sno) as tot
from customer_orders_temp c
join pizza_toppings pt
on c.extras = pt.topping_id
where extras is not NULL
group by 1
order by 2 desc
limit 1;
```

👉 Explanation

- Tables used: **customer_orders_temp**, **pizza_toppings**
- **Count** the **total no. of orders** for each **extra** by using **count** and **group by**

👉 Result

common_extras text	tot bigint
Bacon	4
Cheese	1
Chicken	1

🍕 If a Meat Lovers pizza costs \$12 and Vegetarian costs \$10 and there were no charges for changes -- how much money has Pizza Runner made so far if there are no delivery fees?

👉 Query

```
select sum(case when pizza_id = 1 then 12
else 10
end) as tot_rev
from customer_orders;
```

👉 Explanation

- Table used: **customer_orders**
- **Apply relevant price** for each ordered **pizza** using **case** statement
- **Sum** over the price for **revenue**

👉 Result

tot_rev	
bigint	🔒
	160

Valuable Business Insights 🤖



👉 **Meat Lovers Pizza** is the **most popular** pizza on **Pizza Runner** when compared to Vegetarian Pizza. 🍕 🔥



👉 **1 among 5** customers have ordered **only Vegetarian** Pizzas. 🥕



👉 **Delay of service** for **higher orders** is noticed. 😞



👉 **Revenue generated** without added charges for **extras** is **\$160** and **with extra charges** is **\$166** and **Net Revenue** after runners cut is **\$73.38** 💵



👉 **6PM, 9PM, 11PM, 1PM** are the **happy hours** as most number of orders are placed during these hours. 😄 😄



👉 **Wednesday, Saturday** are the **peak days** for orders followed by Thursday and Friday 🔥



👉 **12 mins** is the **average duration** in mins for the **runners** to **reach Pizza Runner HQ**. 1 2



👉 **1 among 4** runners hold **100% successful** deliveries. 1 2 3 4

Some more Interesting Queries 🔥 🔥

🍕 What was the volume of orders for each day of the week?

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

🍕 Generate an alphabetically ordered comma separated ingredient list for each pizza order from the customer_orders table and add a 2x in front of any relevant ingredients For example: "Meat Lovers: 2xBacon, Beef, ... , Salami"

🍕 If a Meat Lovers pizza was \$12 and Vegetarian \$10 fixed prices with no cost for extras and each runner is paid \$0.30 per kilometre traveled - how much money does Pizza Runner have left over after these deliveries?

For Answers along with explanations: [click here](#) 🖱️