SQL PROJECT :

BUSINESS INSIGHTS FROM PIZZA SALES DATA

rakeshkumargond244@gmail.com

WELCOME TO A DATA-DRIVEN DIVE INTO PIZZA SALES!

EXPLORING ORDER PATTERNS, SALES TRENDS AND CUSTOMER PREFERENCES USING SQL

A STRUCTURED APPROACH TO DATA ANALYSIS WITH 13 BUSINESS-DRIVEN SQL QUERIES

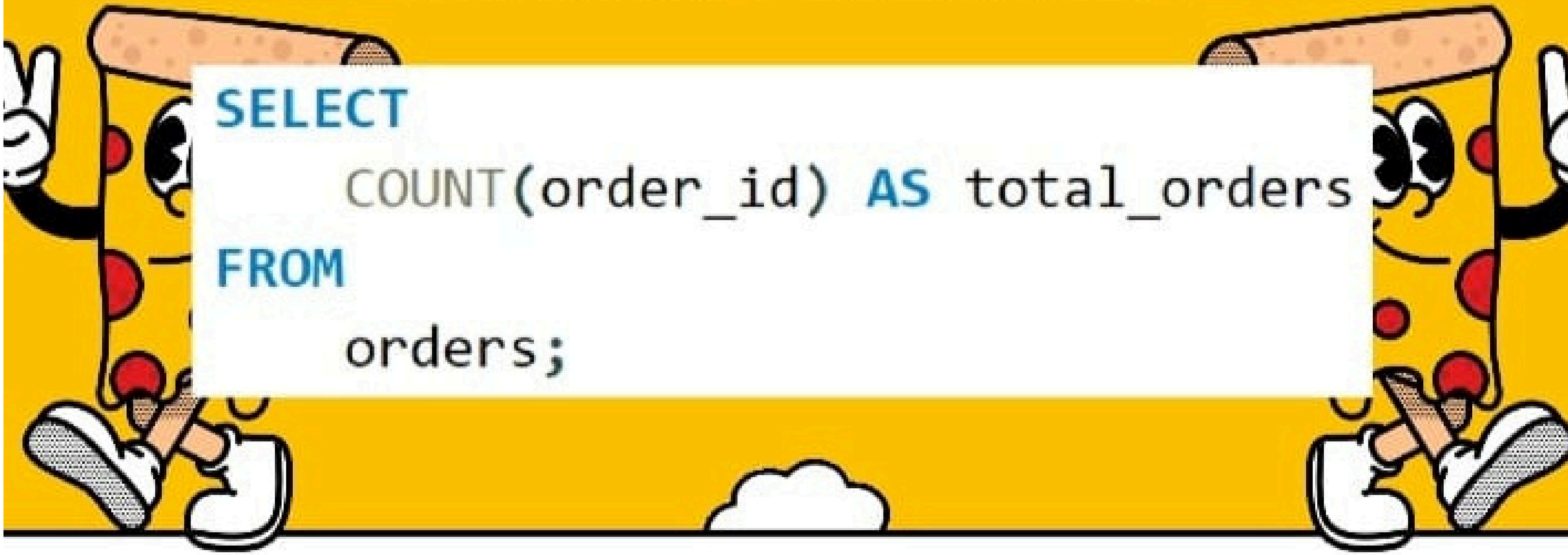
rakeshkumargond244@gmail.com

DATABASE SCHEMA OVERVIEW

(ENTITY-RELATIONSHIP DIAGRAM OF THE PIZZA SALES DATABAS

THIS SCHEMA INCLUDES FOUR INTERRELATED TABLES: ORDERS, ORDER_DETAILS, PIZZAS, AND PIZZA_TYPES.

RETRIEVE THE TOTAL NUMBER OF ORDERS PLACED



```sql
SELECT
    COUNT(order_id) AS total_orders
FROM
    orders;
```

CALCULATE THE TOTAL REVENUE GENERATED FROM PIZZA SALES

```sql
SELECT
    ROUND(SUM(order_details.quantity * pizzas.price),
            2) AS total_sales
FROM
    order_details
        JOIN
    pizzas ON pizzas.pizza_id = order_details.pizza_id
```

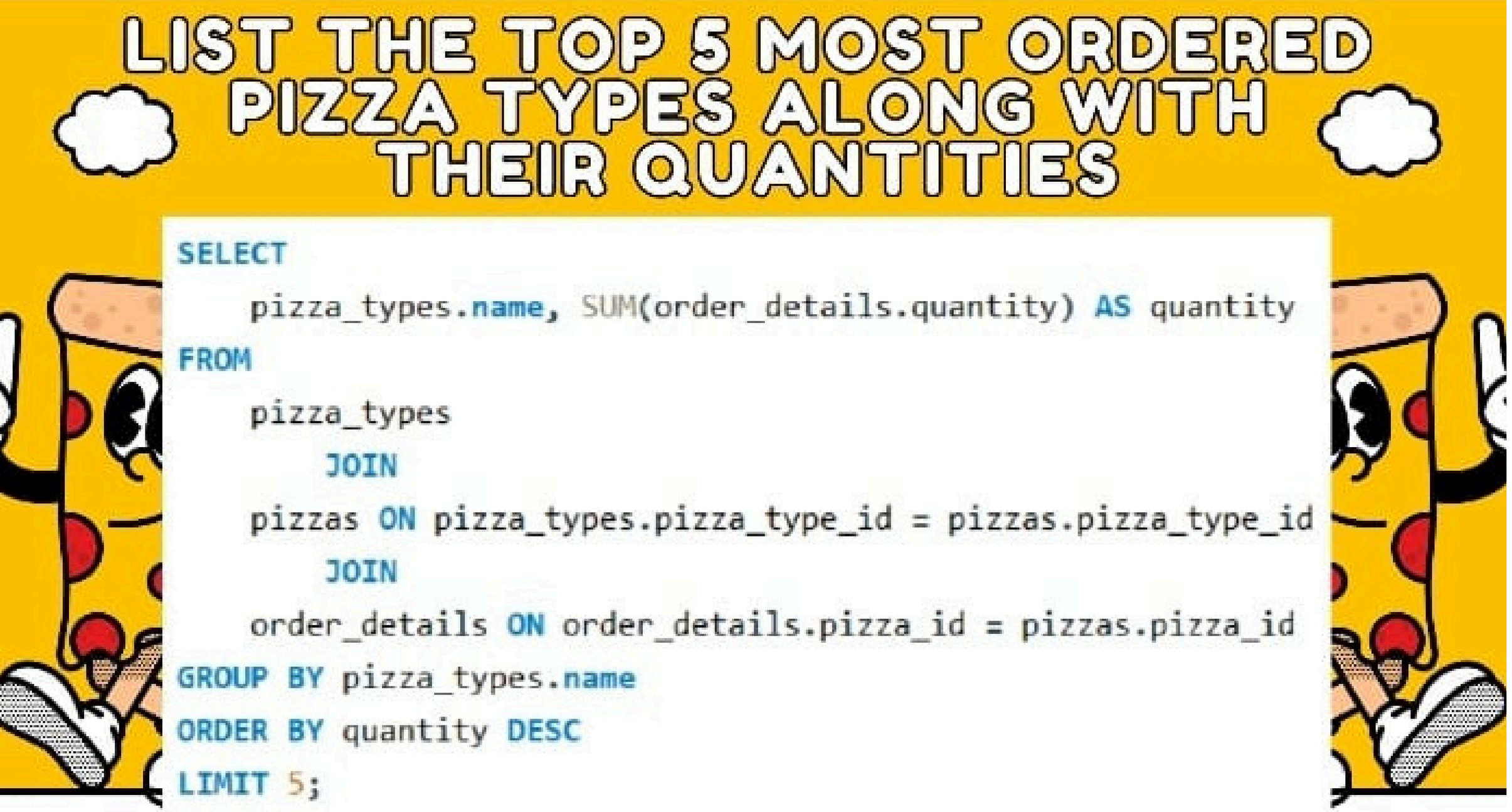
IDENTIFY THE HIGHEST-PRICED PIZZA

```sql
SELECT
    pizza_types.name, pizzas.price
FROM
    pizza_types
        JOIN
    pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
ORDER BY pizzas.price DESC
LIMIT 1;
```

REVENUE GENERATED FROM PIZZA SALES

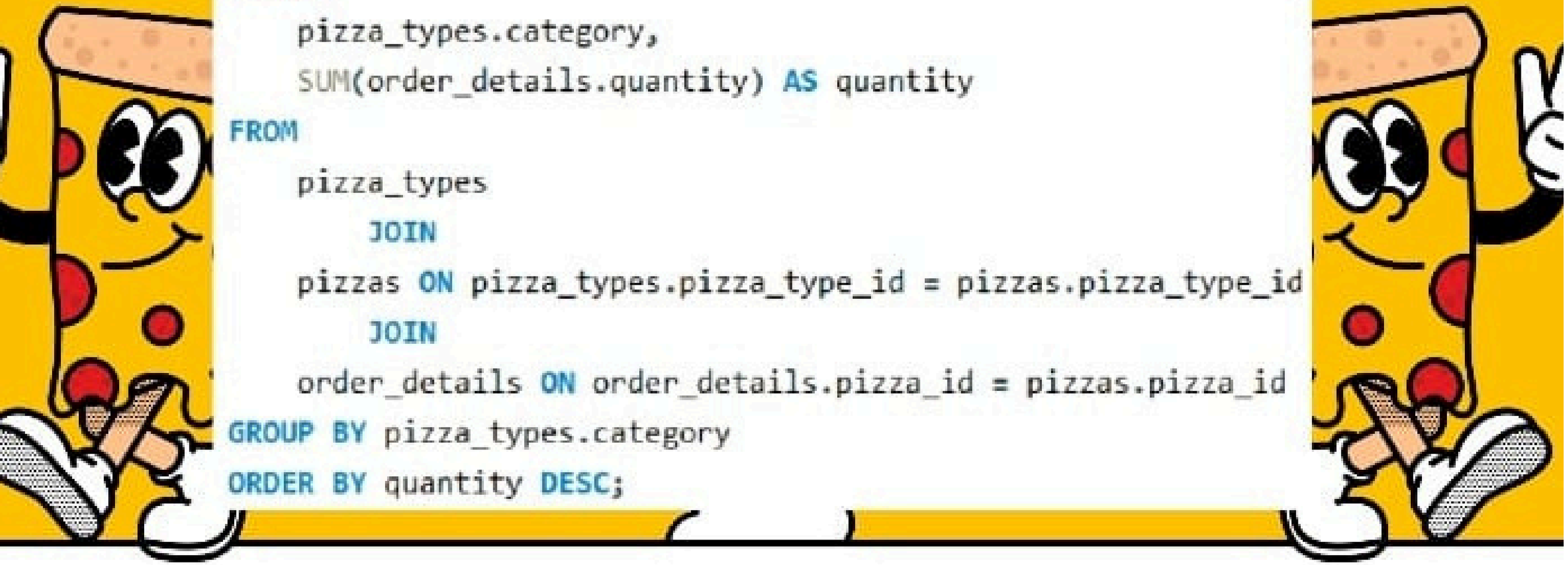
```sql
SELECT
    ROUND(SUM(order_details.quantity * pizzas.price),
            2) AS total_sales
FROM
    order_details
        JOIN
    pizzas ON pizzas.pizza_id = order_details.pizza_id
```

LIST THE TOP 5 MOST ORDERED PIZZA TYPES ALONG WITH THEIR QUANTITIES



```sql
SELECT
    pizza_types.name, SUM(order_details.quantity) AS quantity
FROM
    pizza_types
        JOIN
    pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
        JOIN
    order_details ON order_details.pizza_id = pizzas.pizza_id
GROUP BY pizza_types.name
ORDER BY quantity DESC
LIMIT 5;
```

JOIN THE NECESSARY TABLES TO FIND THE TOTAL QUANTITY OF EACH PIZZA CATEGORY ORDERED

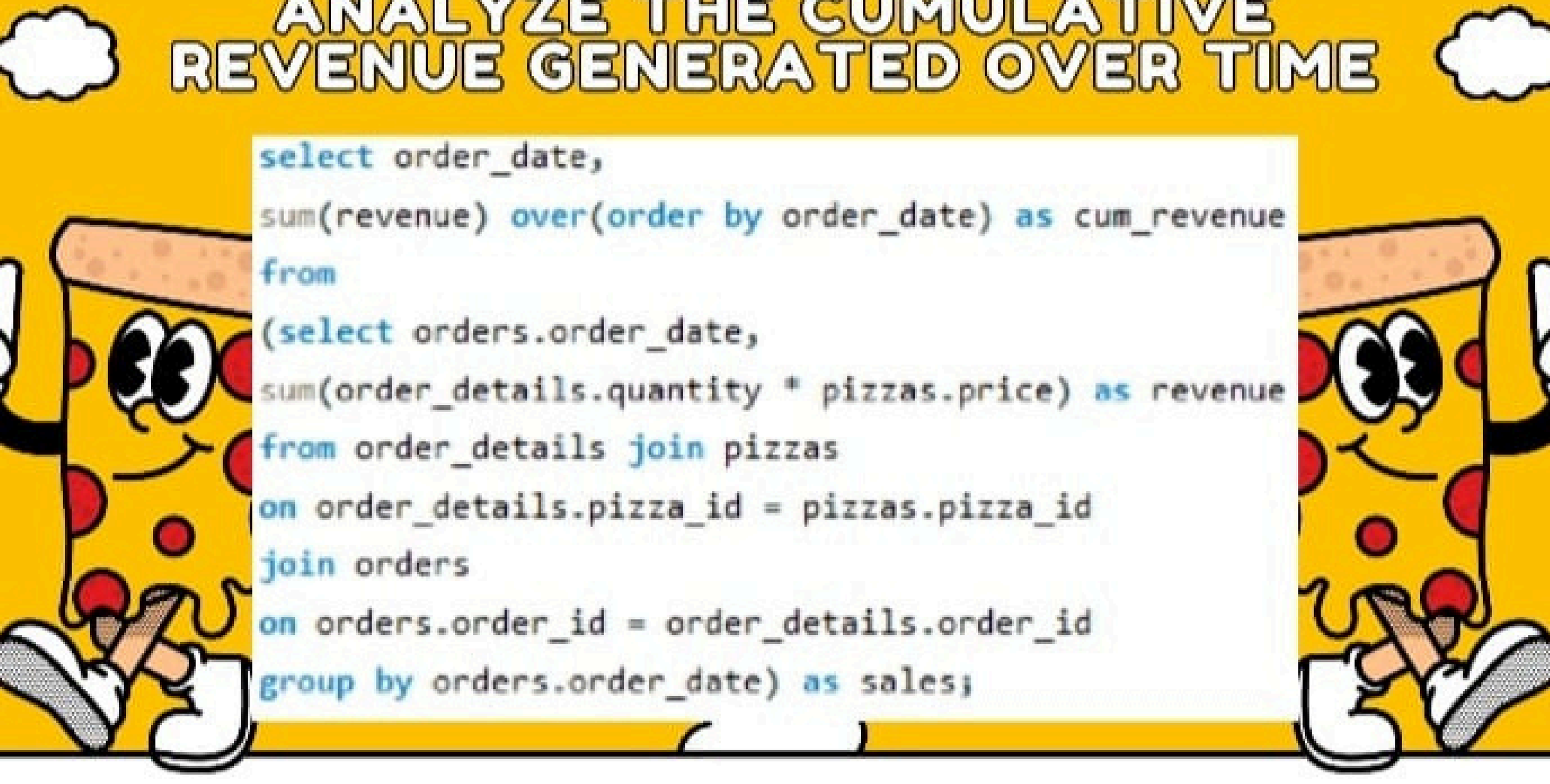


```sql
SELECT
    pizza_types.category,
    SUM(order_details.quantity) AS quantity
FROM
    pizza_types
        JOIN
    pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
        JOIN
    order_details ON order_details.pizza_id = pizzas.pizza_id
GROUP BY pizza_types.category
ORDER BY quantity DESC;
```

THE CATEGORY-WISE DISTRIBUTION OF PIZZAS

```sql
SELECT
    category, COUNT(name)
FROM
    pizza_types
GROUP BY category;
```

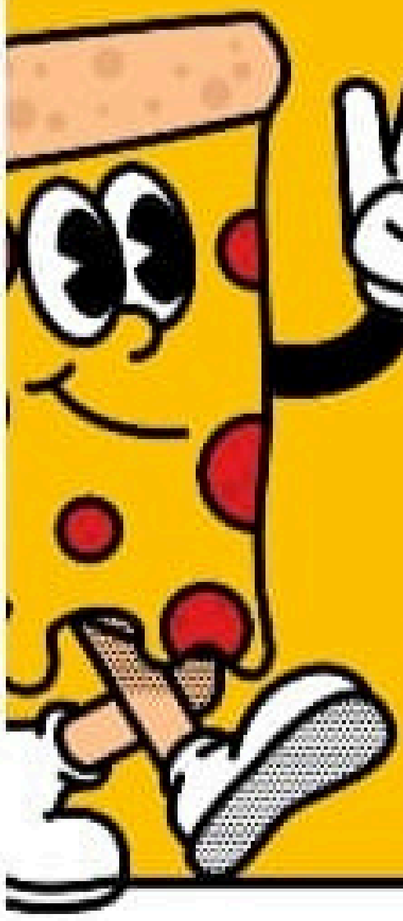
ANALYZE THE CUMULATIVE REVENUE GENERATED OVER TIME



```sql
select order_date,
sum(revenue) over(order by order_date) as cum_revenue
from
(select orders.order_date,
sum(order_details.quantity * pizzas.price) as revenue
from order_details join pizzas
on order_details.pizza_id = pizzas.pizza_id
join orders
on orders.order_id = order_details.order_id
group by orders.order_date) as sales;
```

DETERMINE THE TOP 3 MOST ORDERED PIZZA TYPES BASED ON REVENUE FOR EACH PIZZA CATEGORY

```sql
select name, revenue from
(select category, name, revenue,
rank() over(partition by category order by revenue desc) as rn
from
(select pizza_types.category, pizza_types.name,
sum((order_details.quantity) * pizzas.price) as revenue
from pizza_types join pizzas
on pizza_types.pizza_type_id = pizzas.pizza_type_id
join order_details
on order_details.pizza_id = pizzas.pizza_id
group by pizza_types.category, pizza_types.name) as a) as b
where rn <=3;
```



THANK YOU!