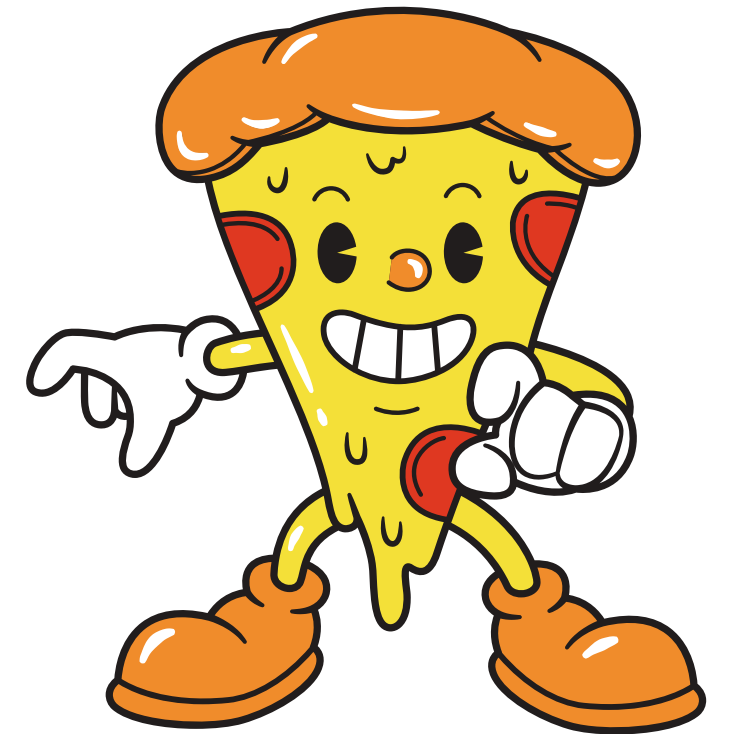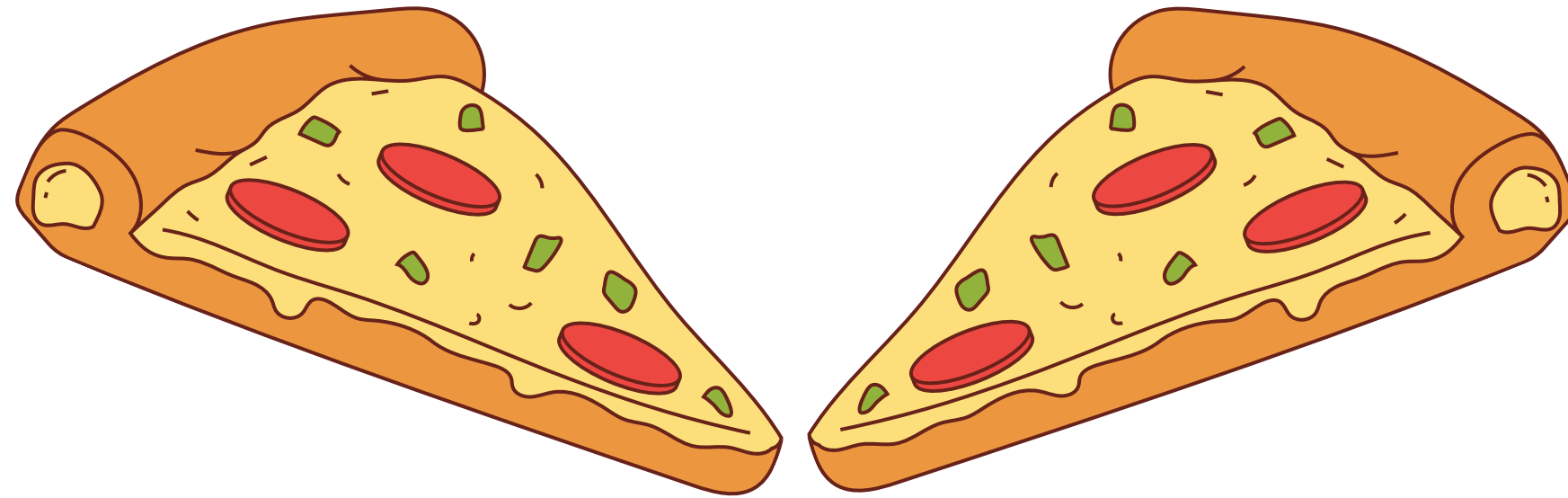# PIZZA HUT

DELICIOUS PIZZA FOR EVERY ONE

HELLO

My name is Abdul Sharukh

My name is Abdul sharukh
This project I have utilize SQL query to solve a questions that for related to pizza sales.

Retrieve the total number of orders placed.

```
Select count(Order_id) as total_orders from orders;
```

	total_orders
▶	21350

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
```

total_sales
817860.05

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;
```

name	price
The Greek Pizza	35.95

Identify the most commom pizza ordered.

```sql
SELECT
    pizzas.size,
    COUNT(Order_details.order_details_id) AS order_count
FROM
    pizzas
        JOIN
    order_details ON pizzas.pizza_id = order_details.pizza_id
GROUP BY pizzas.size
ORDER BY order_count DESC;
```

size	order_count
L	18526
M	15385
S	14137
XL	544
XXL	28

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;
  ```

name	quantity
The Classic Deluxe Pizza	2453
The Barbecue Chicken Pizza	2432
The Hawaiian Pizza	2422
The Pepperoni Pizza	2418
The Thai Chicken Pizza	2371

Join the neccessary 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;
```

	category	quantity
▶	Classic	14888
	Supreme	11987
	Veggie	11649
	Chicken	11050

Determine the distribution of orders by hour of the day.

- ```sql
  SELECT
      HOUR(Order_time) AS hour, COUNT(Order_id) AS order_count
  FROM
      orders
  GROUP BY HOUR(Order_time);
  ```

hour	order_count
11	1231
12	2520
13	2455
14	1472
15	1468
16	1920
17	2336
18	2399
19	2009
20	1642
21	1198
22	663
23	28

Join relevant tables to find the category-wise distribution of pizzas.

- ```sql
  select category,count(name) from pizza_types
  group by category;
  ```

category	count(name)
Chicken	6
Classic	8
Supreme	9
Veggie	9

Group the orders by date and calculate the average number of pizzas ordered per day

```sql
•    SELECT
         ROUND(AVG(quantity), 0) as avg_pizza_ordered_per_day
     FROM
         (SELECT
             orders.order_date, SUM(order_details.quantity) AS quantity
         FROM
             orders
         JOIN order_details ON orders.order_id = order_details.Order_id
         GROUP BY orders.order_date) AS order_quantity;
```

avg_pizza_ordered_per_day
138

Determine the top 3 most ordered pizza types based on revenue.

- ```sql
  select pizza_types.name,
  sum(order_details.quantity*pizzas.price) as revenue
  from pizza_types join pizzas
  on pizzas.pizza_type_id=pizza_types.pizza_type_id
  join order_details
  On order_details.pizza_id=pizzas.pizza_id
  group by pizza_types.name order by revenue desc limit 3;
  ```

name	revenue
▶ The Thai Chicken Pizza	43434.25
The Barbecue Chicken Pizza	42768
The California Chicken Pizza	41409.5

Calculate the percentage contribution of each pizza type to total revenue.

```sql
select pizza_types.category,
round(sum(order_details.quantity*pizzas.price)/(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) *100,2) 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 order by revenue desc;
```

category	revenue
Classic	26.91
Supreme	25.46
Chicken	23.96
Veggie	23.68

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;
```

order_date	cum_revenue
2015-01-01	2713.8500000000004
2015-01-02	5445.75
2015-01-03	8108.15
2015-01-04	9863.6
2015-01-05	11929.55
2015-01-06	14358.5
2015-01-07	16560.7
2015-01-08	19399.05
2015-01-09	21526.4
2015-01-10	23990.350000000002
2015-01-11	25862.65
2015-01-12	27781.7
2015-01-13	29831.300000000003

Determine the top3 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;
```

name	revenue
The Thai Chicken Pizza	43434.25
The Barbecue Chicken Pizza	42768
The California Chicken Pizza	41409.5
The Classic Deluxe Pizza	38180.5
The Hawaiian Pizza	32273.25
The Pepperoni Pizza	30161.75
The Spicy Italian Pizza	34831.25
The Italian Supreme Pizza	33476.75
The Sicilian Pizza	30940.5
The Four Cheese Pizza	32265.70000000065
The Mexicana Pizza	26780.75
The Five Cheese Pizza	26066.5