



PIZZAHUT



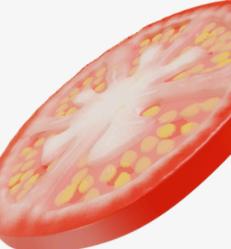


WELCOME TO

PIZZAHUT

PIZZAHUT IS A POPULAR PIZZA RESTAURANT THAT SERVES
A VARIETY OF FRESHLY BAKED PIZZAS, SIDES, AND
BEVERAGES.

IT OPERATES BOTH DINE-IN AND DELIVERY SERVICES,
ENSURING CUSTOMERS CAN ENJOY THEIR FAVORITE
PIZZAS CONVENIENTLY.





VISSION & MISSION

VISSION

TO BECOME THE MOST LOVED PIZZA BRAND BY DELIVERING DELICIOUS, HIGH-QUALITY PIZZAS WITH EXCEPTIONAL SERVICE, USING DATA-DRIVEN INSIGHTS TO ENHANCE CUSTOMER EXPERIENCE AND BUSINESS GROWTH

MISSION

"TO DELIVER HIGH-QUALITY PIZZAS WITH EXCEPTIONAL SERVICE, USING DATA-DRIVEN INSIGHTS TO ENHANCE CUSTOMER EXPERIENCE AND BUSINESS GROWTH."





RETRIEVE THE TOTAL NUMBER OF ORDERS PLACED.

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

Result Grid	
	total_orders
→	21350



CALCULATE THE TOTAL REVENUE GENERATED FROM PIZZA SALES.

```
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.

```
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;
```

The screenshot shows a MySQL Workbench interface with a results grid. The grid has two columns: 'name' and 'price'. A single row is displayed, showing 'The Greek Pizza' in the 'name' column and '35.95' in the 'price' column. The 'Result Grid' tab is selected at the top of the window.

	name	price
▶	The Greek Pizza	35.95

IDENTIFY THE MOST COMMON PIZZA SIZE ORDERED.

```
select quantity, count(order_details_id)
from order_details group by quantity;
```



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

Result Grid | Filter Rows

	size	common_size_orders
▶	XXL	28
	XL	544
	S	14137
	M	15385
	L	18526



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

- **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;
```

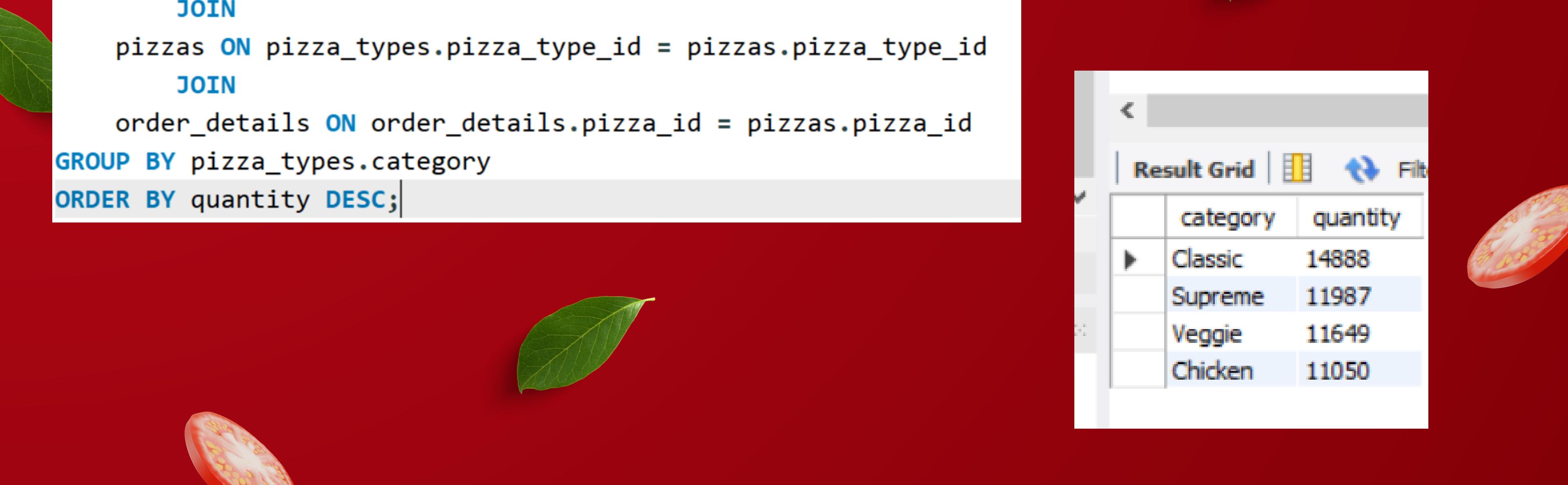
Result Grid | Filter Rows:

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

```
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;
```



A screenshot of a database query results window. The title bar includes 'Result Grid' and a filter icon. The table has two columns: 'category' and 'quantity'. The data shows four rows: Classic (14888), Supreme (11987), Veggie (11649), and Chicken (11050). The Supreme row is highlighted.

	category	quantity
▶	Classic	14888
	Supreme	11987
	Veggie	11649
	Chicken	11050



DETERMINE THE DISTRIBUTION OF ORDERS BY HOUR OF THE DAY.

```
SELECT  
    HOUR(order_time) AS hour, COUNT(order_id) AS order_count  
FROM  
    orders  
GROUP BY HOUR(order_time);
```

Result Grid | Filter Rows:

	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

23	28
10	8
9	1



JOIN RELEVANT TABLES TO FIND THE CATEGORY-WISE DISTRIBUTION OF PIZZAS.

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

Result Grid | Filter Rows:

	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.

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;

Result Grid | Filter Rows

	avg_pizza_ordered_per_day
▶	138



DETERMINE THE TOP 3 MOST ORDERED PIZZA TYPES BASED ON REVENUE.

SELECT

```
    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.name  
ORDER BY revenue DESC  
LIMIT 3;
```

Result Grid | Filter Rows:

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



ANALYZE THE CUMULATIVE REVENUE GENERATED OVER TIME.ALTER

```
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	23990.35000000002
	2015-01-11	25862.65
	2015-01-12	27781.7
	2015-01-13	29831.30000000003

	order_date	cum_revenue
	2015-01-13	29831.30000000003
	2015-01-14	32358.70000000004
	2015-01-15	34343.50000000001
	2015-01-16	36937.65000000001
	2015-01-17	39001.75000000001
	2015-01-18	40978.60000000006
	2015-01-19	43365.75000000001
	2015-01-20	45763.65000000001
	2015-01-21	47804.20000000001
	2015-01-22	50300.90000000001
	2015-01-23	52724.60000000006
	2015-01-24	55013.85000000006
	2015-01-25	56631.40000000001



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

```
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

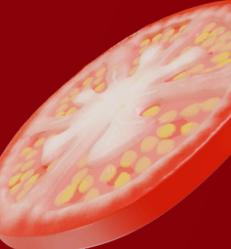


CALCULATE THE PERCENTAGE CONTRIBUTION OF EACH PIZZA TYPE TO TOTAL REVENUE.

```
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
;
```

Result Grid

	category	revenue
▶	Classic	26.91
	Supreme	25.46
	Chicken	23.96
	Veggie	23.68





PIZZAHUT

THANK YOU!

