



Larana, Inc.

TASTE
THE
BEST
FROM
THE
OVEN

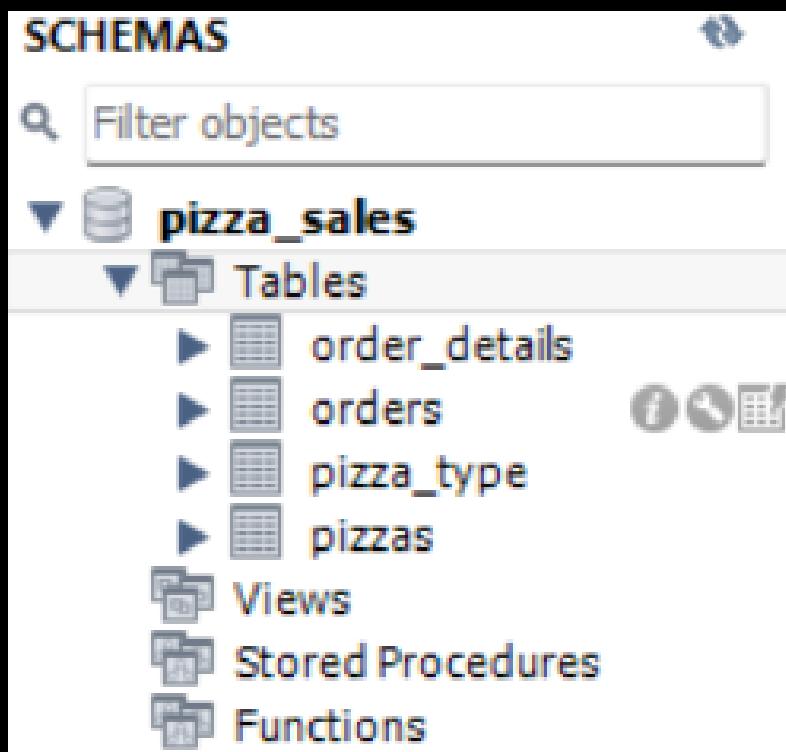


+123-456-7890
@reallygreatsite
123 Anywhere St., Any City, ST 12345

MYSQL Project

Pizza Sales

By using MYSQL I find some insightful result , data management,query Optimization.



-- Retrieve the total number
of orders placed.

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

	COUNT(order_id)
▶	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 order_details.pizza_id = pizzas.pizza_id;
```

	total_sales
▶	817860.05

-- Identify the highest-priced pizza.

```
SELECT
    pizza_type.name, pizzas.price
FROM
    pizza_type
    join pizzas
    On pizza_type.pizza_type_id = pizzas.pizza_type_id
ORDER BY price DESC
LIMIT 1;
```

	name	price
▶	The Greek Pizza	35.95

-- Identify the most common pizza size ordered.

```
SELECT
    pizzas.size, COUNT(order_details.order_details_id)
FROM
    pizzas
    JOIN
        order_details ON pizzas.pizza_id = order_details.pizza_id
GROUP BY pizzas.size
ORDER BY COUNT(order_details.order_details_id) DESC
LIMIT 1;
```

	size	COUNT(order_details.order_details_id)
▶	L	18526

-- List the top 5 most ordered pizza types along with their quantities.

```
SELECT pizzas.size, COUNT(order_details.quantity) AS quantity
FROM pizzas
    JOIN order_details ON pizzas.pizza_id = order_details.pizza_id
GROUP BY pizzas.size
ORDER BY pizzas.size
LIMIT 5;
```

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

-- Join the necessary tables to find the total quantity of each pizza category ordered.

```
SELECT
```

```
    pizza_type.category, SUM(order_details.quantity)
        as total_quantity
FROM
    pizza_type
        JOIN
    pizzas ON pizza_type.pizza_type_id = pizzas.pizza_type_id
        JOIN
    order_details ON pizzas.pizza_id = order_details.pizza_id
GROUP BY pizza_type.category
ORDER BY SUM(order_details.quantity);
```

	category	total_quantity
▶	Chicken	11050
	Veggie	11649
	Supreme	11987
	Classic	14888

Determine the distribution of orders by hour of the day.

```
SELECT
```

```
    HOUR(time), COUNT(order_id)
```

```
FROM
```

```
orders
```

```
GROUP BY HOUR(time);
```

	HOUR(time)	COUNT(order_id)
▶	11	1231
	12	2520
	13	2455
	14	1472
	15	1468
	16	1920
	17	2336
	18	2399

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

```
SELECT  
    category, COUNT(name)  
FROM  
    pizza_type  
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.

```
SELECT
    AVG(quantity)
FROM
    (SELECT
        orders.date, SUM(order_details.quantity) AS quantity
    FROM
        orders
    JOIN order_details ON orders.order_id = order_details.order_id
    GROUP BY orders.date) AS order_quantity ;
```

	Avg(quantity)
▶	138.4749

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

```
SELECT
    pizza_type.name,
    SUM(pizzas.price * order_details.quantity) AS revenue
FROM
    pizza_type
        JOIN
    pizzas ON pizza_type.pizza_type_id = pizzas.pizza_type_id
        JOIN
    order_details ON order_details.pizza_id = pizzas.pizza_id
GROUP BY pizza_type.name
LIMIT 3;
```

	name	revenue
▶	The Hawaiian Pizza	32273.25
	The Classic Deluxe Pizza	38180.5
	The Five Cheese Pizza	26066.5

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

SELECT

```
    pizza_type.category,  
    ROUND(SUM(order_details.quantity * pizzas.price) / (SELECT  
        SUM(order_details.quantity * pizzas.price)  
        AS total_sales  
    FROM  
        order_details  
        JOIN  
        pizzas ON order_details.pizza_id = pizzas.pizza_id) * 100,  
    2) AS revenue  
FROM  
    pizza_type  
    JOIN  
    pizzas ON pizzas.pizza_type_id = pizza_type.pizza_type_id  
    JOIN  
    order_details ON order_details.pizza_id = pizzas.pizza_id  
    GROUP BY pizza_type.category;
```

	category	revenue
▶	Classic	26.91
	Veggie	23.68
	Supreme	25.46
	Chicken	23.96

-- Analyze the cumulative revenue generated over time.

```
SELECT
```

```
    orders.date,  
    ROUND(SUM(pizzas.price * order_details.quantity),  
         2) AS revenue
```

```
FROM
```

```
order_details
```

```
    JOIN
```

```
pizzas ON pizzas.pizza_id = order_details.pizza_id
```

```
    JOIN
```

```
orders ON orders.order_id = order_details.order_id
```

```
GROUP BY orders.date;
```

	date	revenue
▶	01-01-2015	2713.85
	02-01-2015	2731.9
	03-01-2015	2662.4
	04-01-2015	1755.45
	05-01-2015	2065.95
	06-01-2015	2428.95
	07-01-2015	2202.2
	08-01-2015	2838.35

-- Determine the top 3 most ordered pizza types based on revenue for each pizza category

```
SELECT category, name, revenue
FROM (
    SELECT
        pizza_type.category,
        pizza_type.name,
        SUM(order_details.quantity * pizzas.price) AS revenue,
        RANK() OVER (PARTITION BY pizza_type.category
                     ORDER BY SUM(order_details.quantity * pizzas.price) DESC) AS rn
    FROM pizzas
    JOIN pizza_type ON pizza_type.pizza_type_id = pizzas.pizza_type_id
    JOIN order_details ON order_details.pizza_id = pizzas.pizza_id
    GROUP BY pizza_type.category, pizza_type.name
) AS ranked_pizzas
WHERE rn <= 3;
```

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