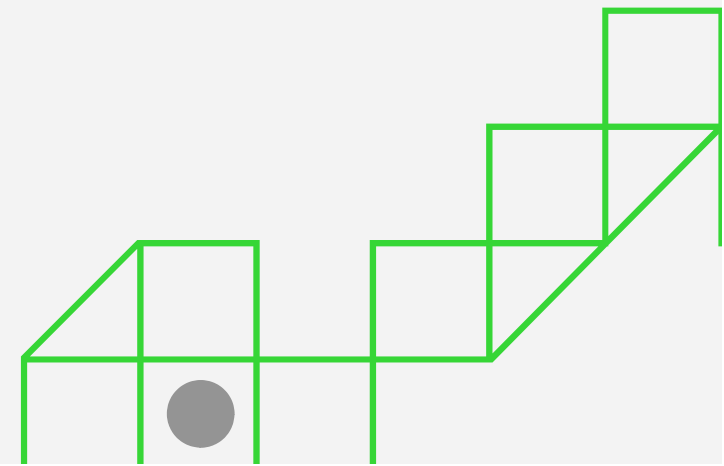
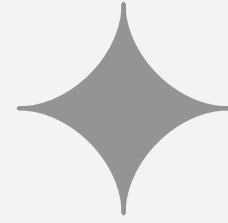




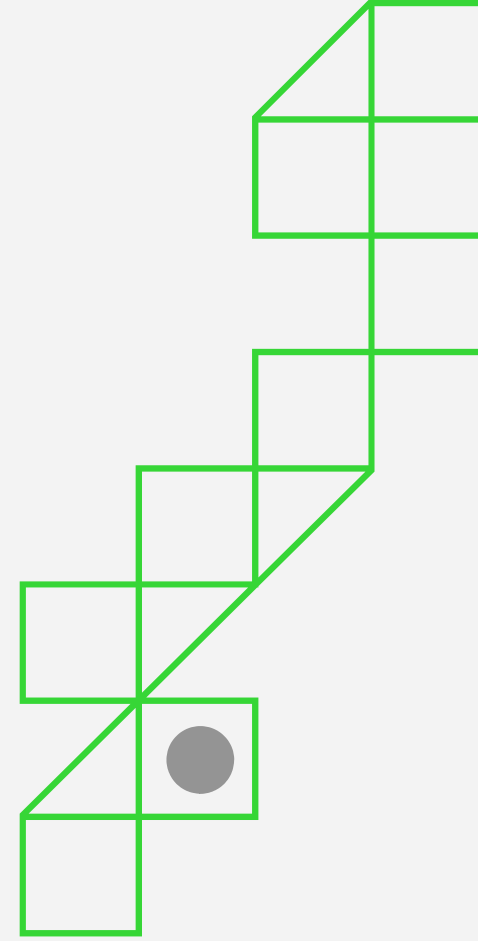
Analyzing Pizza Sales Trends with SQL: A Data-Driven Approach



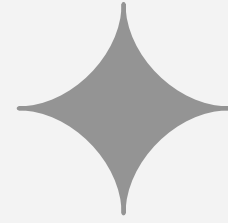


Introduction

Welcome to **Analyzing Pizza Sales Trends** with SQL. In this presentation, we will explore the **data-driven** approach to understanding pizza sales. We will analyze trends and make informed decisions based on real-time data.



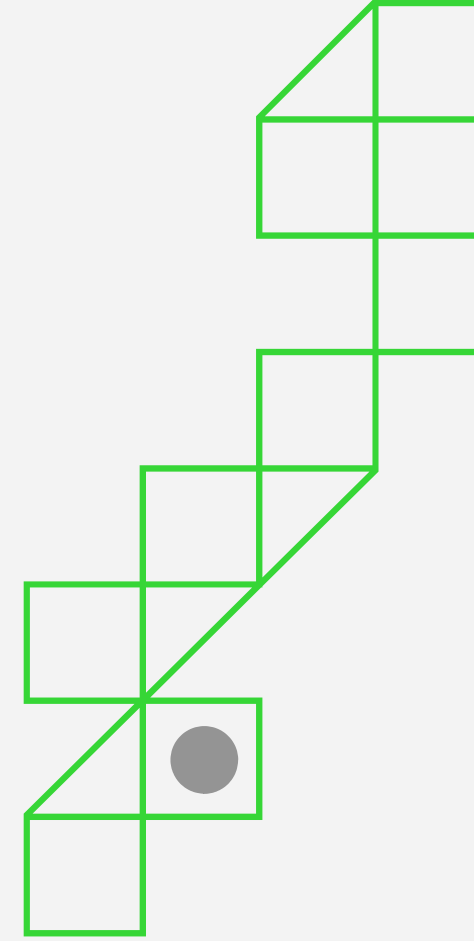
Understanding Data



Data Overview We have a database with four tables, each containing 1,000 entries:

1. `order_details`: Tracks order specifics (`order_id`, `pizza_id`, `quantity`, `price`).
2. `orders`: Contains order summaries (`order_id`, `customer_id`, `order_date`).
3. `pizza_types`: Describes pizza varieties (`pizza_type_id`, `name`, `category`, `ingredients`).
4. `pizzas`: Lists individual pizzas (`pizza_id`, `pizza_type_id`, `size`, `price`).

These tables together provide insights into orders, customer preferences, and pizza details.



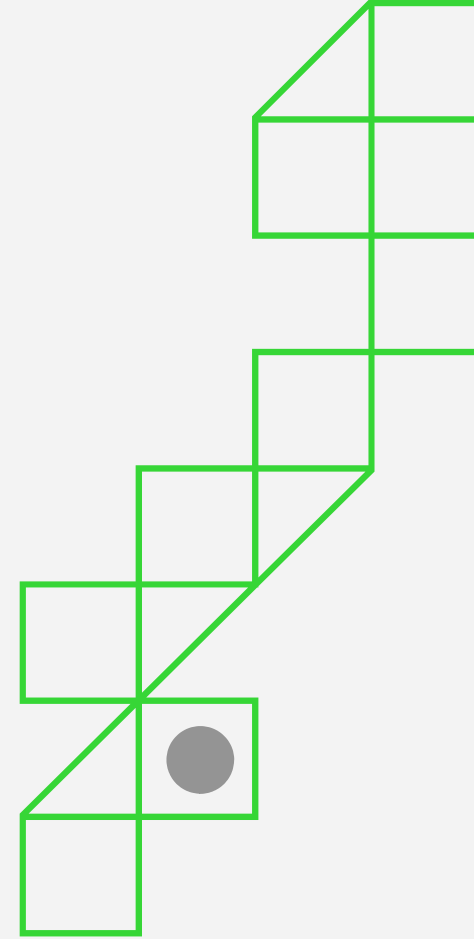
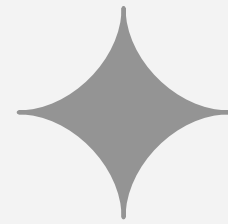
Total no. of Orders Placed

Query

```
-- Retrive the total number of orders placed
SELECT
    COUNT(order_id) AS total_orders
FROM
    orders;
```

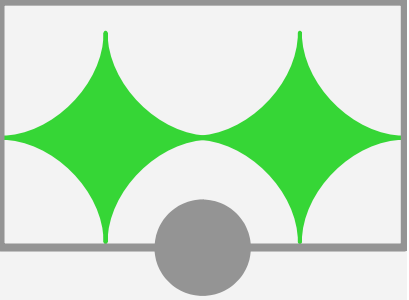
Output

	total_orders
▶	11425





Total revenue generated



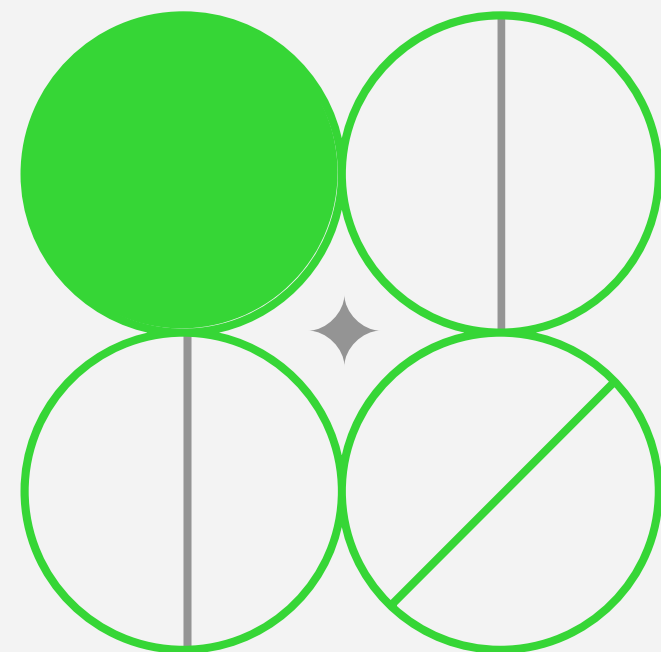
Query

```
-- calculate the total revenue generated from pizza sales

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

Output

	total_sales
▶	129973.3





Most expensive pizza

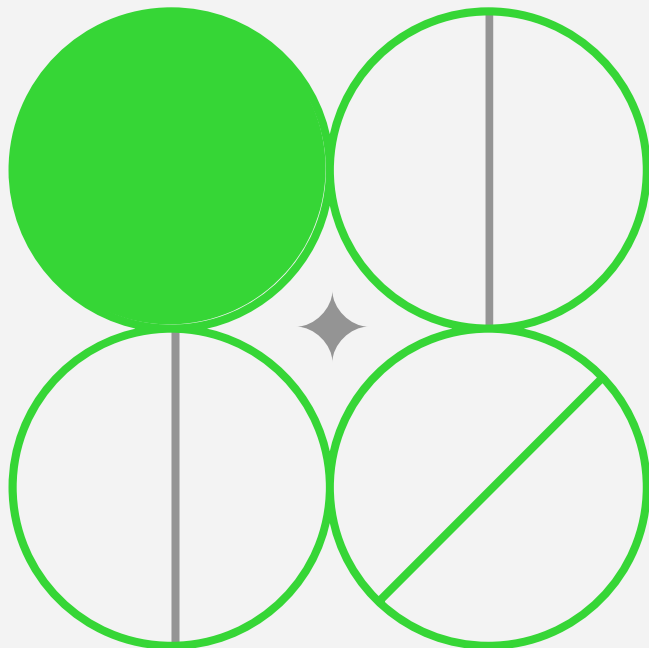
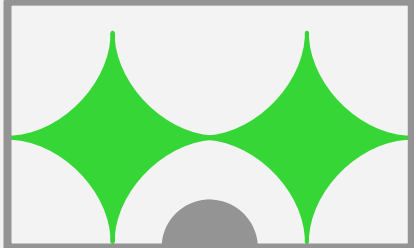
Query

```
-- Identify the highest priced pizza

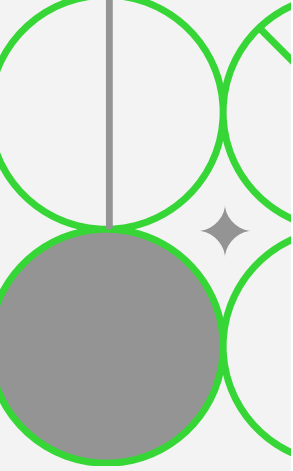
SELECT
    pizza_types.name, pizzas.price
FROM
    pizza_types
    JOIN
        pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
ORDER BY pizzas.price DESC
LIMIT 1;
```

Output

	name	price
▶	The Greek Pizza	35.95



The most common pizza sizes ordered

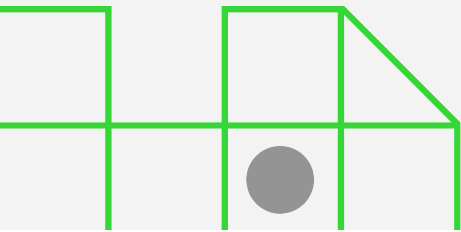
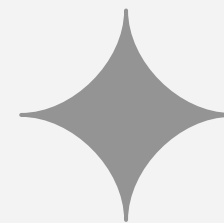


Query

```
-- identify the most common pizza size ordered
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;
```

Output

	size	order_count
▶	L	2976
	M	2389
	S	2294
	XL	84
	XXL	5



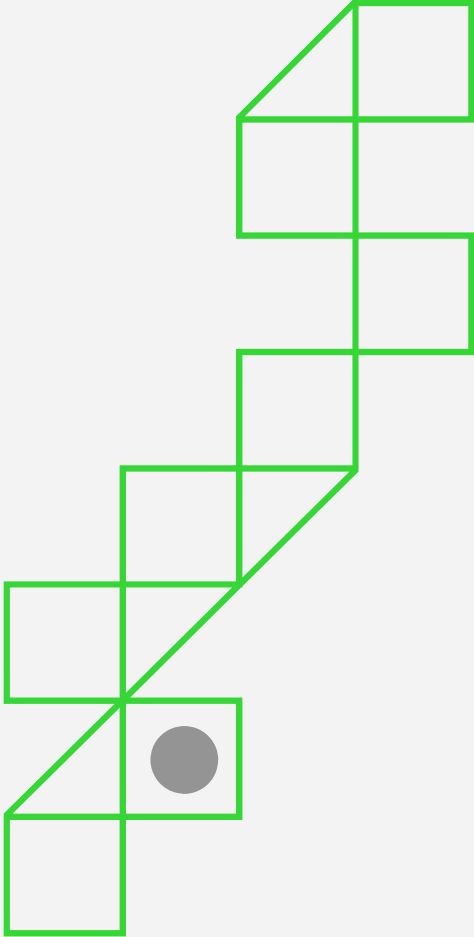
5 most ordered pizza types

Query

```
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;
```

Output

	name	quantity
►	The Pepperoni Pizza	427
	The Barbecue Chicken Pizza	391
	The California Chicken Pizza	389
	The Hawaiian Pizza	369
	The Thai Chicken Pizza	356



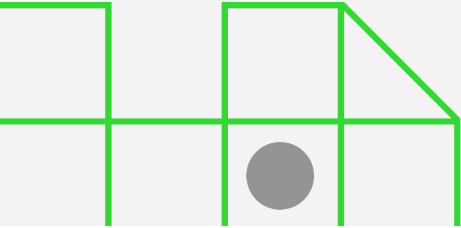
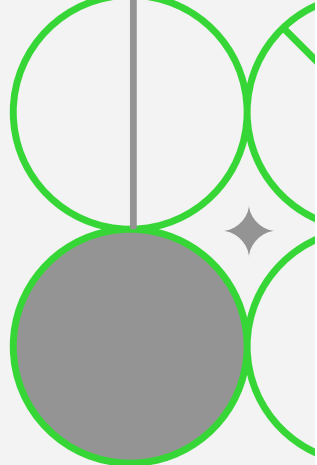
Total quantity of each pizza category ordered

Query

```
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;
```

Output

	category	quantity
▶	Classic	2348
	Supreme	1935
	Veggie	1880
	Chicken	1727



Which hour has the highest number of orders

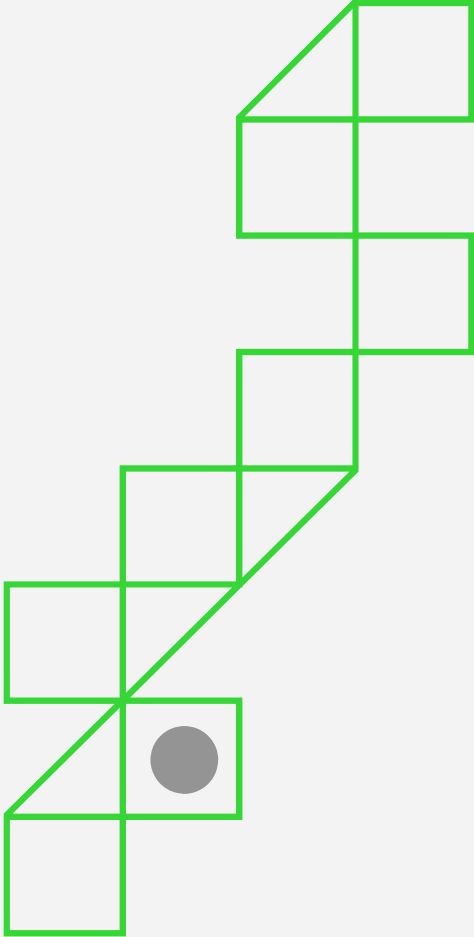


Query

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

Output

	hour	order_count
▶	11	636
	12	1375
	13	1276
	14	841
	15	786
	16	1014
	17	1309
	18	1258
	19	1061
	20	884
	21	631
	22	334
	23	14
	10	6





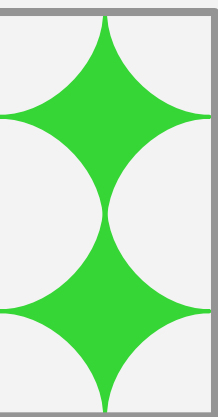
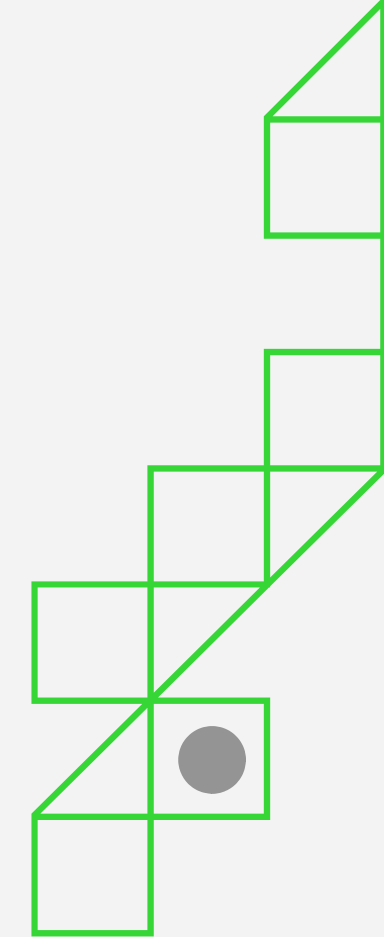
Average number of pizzas ordered per day

Query

```
SELECT
    AVG(quantity) AS avg_pizza_ordered_per_day
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;
```

Output

	avg_pizza_ordered_per_day
▶	136.0345



Which pizza category produces the most revenue

Query

```
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;
```

Output

	name	revenue
▶	The Barbecue Chicken Pizza	6957.25
	The California Chicken Pizza	6711.75
	The Thai Chicken Pizza	6483

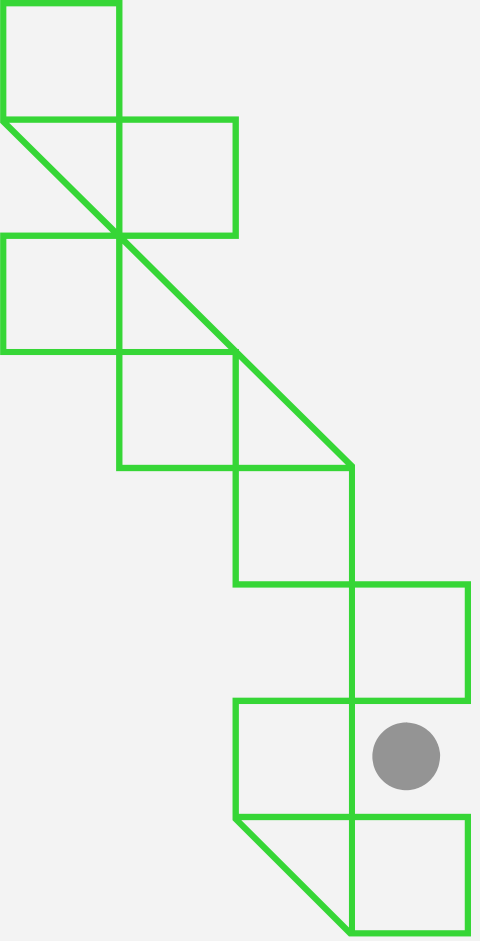
Percentage contribution of each pizza types to the total revenue

Query

```
SELECT
  pizza_types.category,
  (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 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;
```

Output

	category	revenue
▶	Classic	26.665284331474233
	Supreme	25.69100730688538
	Veggie	24.125608875053647
	Chicken	23.51809948658686



Thanks!

