

A close-up photograph of a hand using a pizza cutter to lift a slice of pizza. The pizza is topped with melted cheese, sliced cherry tomatoes, black olives, and fresh herbs. The background is dark and slightly blurred, showing more of the pizza and some whole tomatoes.

Pizza Sale Data Analysis Project

Name- Akash Gunjal



Introduction

- This project is to design the DB Tables, load the data into DB and to solve the questions related data analysis of pizza sale using sql queries.
- Also visualize in power BI where It help to understand the insights of the data very clearly.

```
1 • SELECT * FROM pizzhut.pizza_types;
```

pizza_type_id	name	category	ingredients
bbq_ckn	The Barbecue Chicken Pizza	Chicken	Barbecued Chicken, Red Pepper
cali_ckn	The California Chicken Pizza	Chicken	Chicken, Antichoke, Spinach, Ga
ckn_alfredo	The Chicken Alfredo Pizza	Chicken	Chicken, Red Onions, Red Pepp
ckn_pesto	The Chicken Pesto Pizza	Chicken	Chicken, Tomatoes, Red Pepp
southw_ckn	The Southwest Chicken Pizza	Chicken	Chicken, Tomatoes, Red Pepp
thai_ckn	The Thai Chicken Pizza	Chicken	Chicken, Pineapple, Tomatoes, ...
big_meat	The Big Meat Pizza	Classic	Bacon, Pepperoni, Italian Sausa
classic_dlx	The Classic Deluxe Pizza	Classic	Pepperoni, Mushrooms, Red On
hawaiian	The Hawaiian Pizza	Classic	Sliced Ham, Pineapple, Mozzare
ital_cpdllo	The Italian Capocollo Pizza	Classic	Capocollo, Red Peppers, Tomat
napolitana	The Neapolitan Pizza	Classic	Tomatoes, Anchovies, Green Oli
pep_msh_pesto	The Pepperoni, Mushroom, a	Classic	Pepperoni, Mushrooms, Green
pepperoni	The Pepperoni Pizza	Classic	Mozzarella Cheese, Pepperoni
the_greek	The Greek Pizza	Classic	Kalamata Olives, Feta Cheese
brin_cane	The Brin Cane Pizza	Supreme	Brie Cheese, Prosciutto, C
calabrese	The Calabrese Pizza	Supreme	Nduja Salami, Pancetta, Tomat
ital_sune	The Italian Sun-dried Tomato	Supreme	Calabrese Salami, Cannellini To

```
1 • SELECT * FROM orders;
```

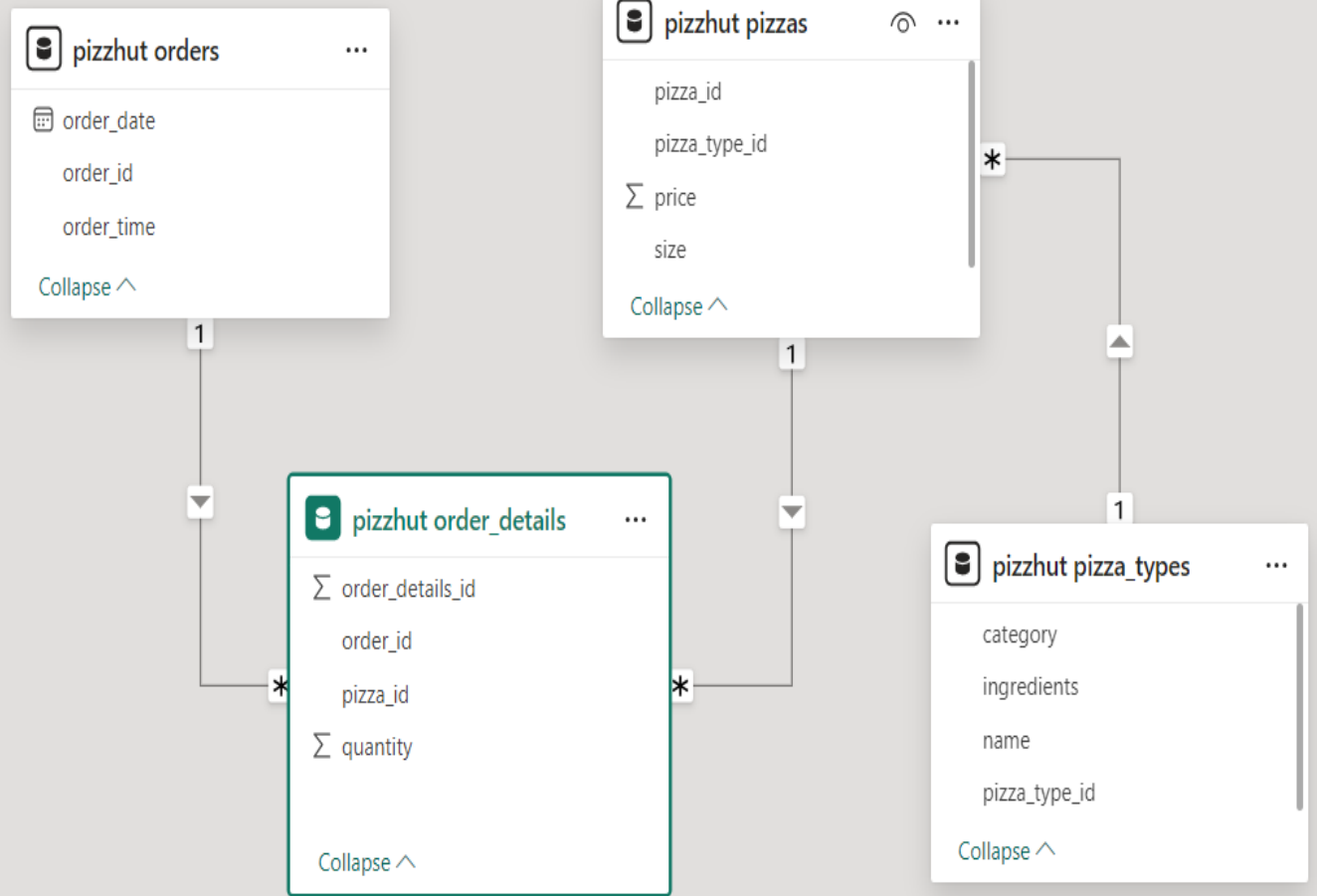
order_id	order_date	order_time
1	2015-01-01	11:38:36
2	2015-01-01	11:57:40
3	2015-01-01	12:12:28
4	2015-01-01	12:16:31
5	2015-01-01	12:21:30
6	2015-01-01	12:29:36
7	2015-01-01	12:50:37
8	2015-01-01	12:51:37
9	2015-01-01	12:52:01
10	2015-01-01	13:00:15
11	2015-01-01	13:02:59
12	2015-01-01	13:04:41
13	2015-01-01	13:11:55
14	2015-01-01	13:14:19
15	2015-01-01	13:33:00
16	2015-01-01	13:34:07
17	2015-01-01	13:53:00

```
1 • SELECT * FROM pizzhut.order_details;
```

order_details_id	order_id	pizza_id	quantity
1	1	hawaiian_m	1
2	2	classic_dlx_m	1
3	2	five_cheese_l	1
4	2	ital_supr_l	1
5	2	mexicana_m	1
6	2	thai_ckn_l	1
7	3	ital_supr_m	1
8	3	prsc_argla_l	1
9	4	ital_supr_m	1
10	5	ital_supr_m	1
11	6	bbq_ckn_s	1
12	6	the_greek_s	1
13	7	spinach_sup...	1
14	8	spinach_sup...	1
15	9	classic_dlx_s	1

```
1 • SELECT * FROM pizzas;
```

pizza_id	pizza_type_id	size	price
bbq_ckn_s	bbq_ckn	S	12.75
bbq_ckn_m	bbq_ckn	M	16.75
bbq_ckn_l	bbq_ckn	L	20.75
cali_ckn_s	cali_ckn	S	12.75
cali_ckn_m	cali_ckn	M	16.75
cali_ckn_l	cali_ckn	L	20.75
ckn_alfredo_s	ckn_alfredo	S	12.75
ckn_alfredo_m	ckn_alfredo	M	16.75
ckn_alfredo_l	ckn_alfredo	L	20.75
ckn_pesto_s	ckn_pesto	S	12.75
ckn_pesto_m	ckn_pesto	M	16.75
ckn_pesto_l	ckn_pesto	L	20.75
southw_ckn_s	southw_ckn	S	12.75
southw_ckn_m	southw_ckn	M	16.75
southw_ckn_l	southw_ckn	L	20.75
thai_ckn_s	thai_ckn	S	12.75





1) Retrieve the total number of orders placed.

```
1  -- 1) Retrieve the total number of orders placed.
2
3  • select count(order_id) as total_orders from orders;
4
5
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content

total_orders
21350

2) Calculate the total revenue generated from pizza sales.

```
1  -- 2) Calculate the total revenue generated from pizza sales
2
3  • SELECT
4      ROUND(SUM(order_details.quantity * pizzas.price),
5             2) AS total_sales
6  FROM
7      order_details
8  JOIN
9      pizzas ON pizzas.pizza_id = order_details.pizza_id
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content

total_sales
817860.05



3) Identify the highest-priced pizza.

```
1  -- 3)Identify the highest-priced pizza.
2
3  • SELECT
4      pizza_types.name, pizzas.price
5  FROM
6      pizza_types
7      JOIN
8      pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
9  ORDER BY pizzas.price DESC
10 LIMIT 1;
```

Result Grid

	name	price
▶	The Greek Pizza	35.95

4) Identify the most common pizza size ordered.

```
1  -- 4)Identify the most common pizza size ordered.
2
3  • SELECT
4      pizzas.size,
5      COUNT(order_details.order_details_id) AS order_count
6  FROM
7      pizzas
8      JOIN
9      order_details ON pizzas.pizza_id = order_details.pizza_id
10 GROUP BY pizzas.size
11 ORDER BY order_count DESC;
```

Result Grid

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



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

```
1  -- 5) List the top 5 most ordered pizza types along with their quantities
2
3  SELECT
4      pizza_types.name, SUM(order_details.quantity) AS quantity
5  FROM
6      pizza_types
7      JOIN
8      pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
9      JOIN
10     order_details ON order_details.pizza_id = pizzas.pizza_id
11  GROUP BY pizza_types.name
12  ORDER BY quantity DESC
13  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

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

```
1  -- 6) Join the necessary tables to find the total quantity
2  -- of each pizza category ordered.
3  SELECT
4      pizza_types.category,
5      SUM(order_details.quantity) AS quantity
6  FROM
7      pizza_types
8      JOIN
9      pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
10     JOIN
11     order_details ON order_details.pizza_id = pizzas.pizza_id
12  GROUP BY pizza_types.category
13  ORDER BY quantity DESC;
```

category	quantity
Classic	14888
Supreme	11987
Veggie	11649
Chicken	11050



7) Determine the distribution of orders by hour of the day.

```
1  -- 7) Determine the distribution of orders by hour of the day
2
3  • SELECT
4    HOUR(order_time), COUNT(order_id)
5  FROM
6    orders
7  GROUP BY HOUR(order_time);
```

HOUR(order_time)	COUNT(order_id)
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
10	8
9	1

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

```
1  -- 8) Join relevant tables to find the category-wise distribution of pizzas
2
3  • SELECT
4    category, COUNT(name)
5  FROM
6    pizza_types
7  GROUP BY category;
```

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



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

```
1  -- 9)Group the orders by date and calculate the average number of pizzas ordered per day
2
3  • SELECT
4      ROUND(AVG(quantity), 0) AS avg_pizza_ordered_per_day
5  FROM
6      (SELECT
7          orders.order_date, SUM(order_details.quantity) AS quantity
8      FROM
9          orders
10         JOIN order_details ON orders.order_id = order_details.order_id
11        GROUP BY orders.order_date) AS order_quantity
```

Result Grid

avg_pizza_ordered_per_day
138

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

```
1  -- 10)Determine the top 3 most ordered pizza types based on revenue.
2
3  • SELECT
4      pizza_types.name,
5      SUM(order_details.quantity * pizzas.price) AS revenue
6  FROM
7      pizza_types
8      JOIN
9      pizzas ON pizzas.pizza_type_id = pizza_types.pizza_type_id
10     JOIN
11     order_details ON order_details.pizza_id = pizzas.pizza_id
12 GROUP BY pizza_types.name
13 ORDER BY revenue DESC limit 3;
```

Result Grid

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



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

```
1  -- 11) Calculate the percentage contribution of each pizza type to total revenue.
2  * SELECT
3      pizza_types.category,
4      ROUND(SUM(order_details.quantity * pizzas.price) / (SELECT
5          ROUND(SUM(order_details.quantity * pizzas.price),
6              2) AS sales
7      FROM
8          order_details
9          JOIN
10             pizzas ON order_details.pizza_id = pizzas.pizza_id) * 100,
11          2) AS revenue
12  FROM
13      pizza_types
14      JOIN
15          pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
16      JOIN
17          order_details ON order_details.pizza_id = pizzas.pizza_id
18  GROUP BY pizza_types.category
19  ORDER BY revenue DESC
```

category	revenue
Classic	26.91
Supreme	25.46
Chicken	23.96
Veggie	23.68

12) Analyze the cumulative revenue generated over time.

```
1  -- 12) Analyze the cumulative revenue generated over time.
2
3  * select order_date,
4      sum(revenue) over (order by order_date) as cum_revenue
5  from
6      (select orders.order_date, sum(order_details.quantity * pizzas.price) as revenue
7      from order_details join pizzas on order_details.pizza_id = pizzas.pizza_id
8      join orders
9      on orders.order_id = order_details.order_id
10     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



13) Determine the top 3 most ordered pizza types based on revenue for each pizza category.

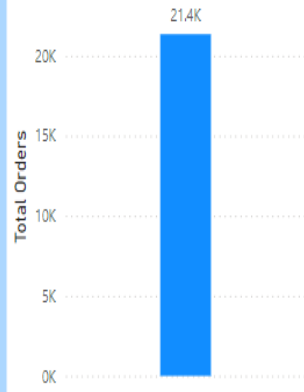
```
1  -- 13)Determine the top 3 most ordered pizza types based on revenue for each pizza category.
2
3  • select name,revenue from
4  (select category, name, revenue,
5   rank() over(partition by category order by revenue desc) as rn
6   from
7   (select pizza_types.category,pizza_types.name,
8    sum(order_details.quantity*pizzas.price)as revenue
9    from pizza_types join pizzas
10   on pizza_types.pizza_type_id = pizzas.pizza_type_id
11   join order_details
12   on order_details.pizza_id = pizzas.pizza_id
13   group by pizza_types.category, pizza_types.name) as a) as b
14  where rn <= 3;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: |

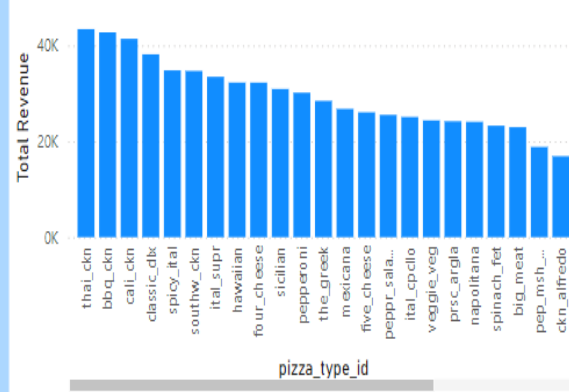
	name	revenue
▶	The Thai Chicken Pizza	43434.25
	The Barbecue Chicken Pizza	42768
	The California Chicken Pizza	41409.5
	The Classic Deluxe Pizza	38180.5
	The Hawaiian Pizza	32273.25
	The Pepperoni Pizza	30161.75
	The Spicy Italian Pizza	34831.25
	The Italian Supreme Pizza	33476.75
	The Sicilian Pizza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



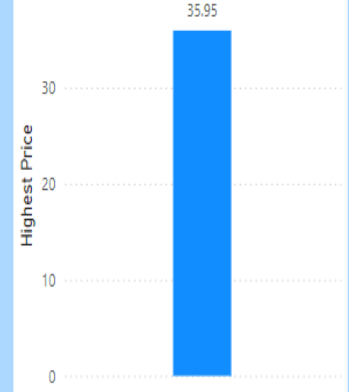
Total Orders



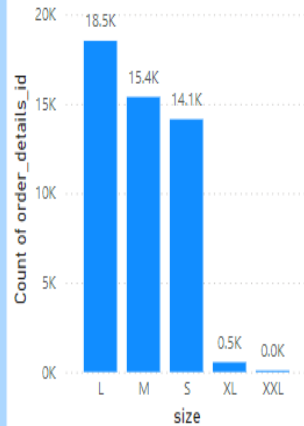
total revenue generated



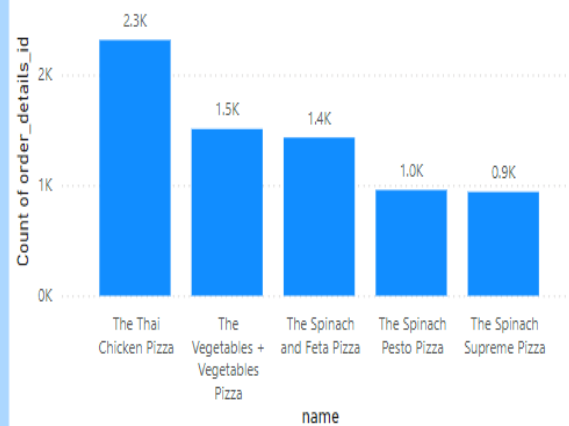
Highest Price pizza



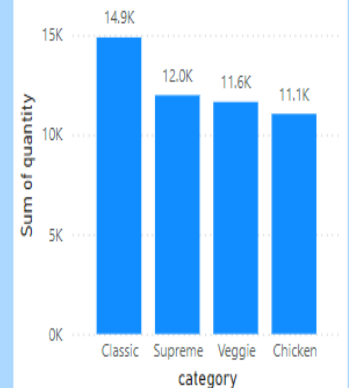
Most common pizza size ordered



most ordered pizza types along with their quantities

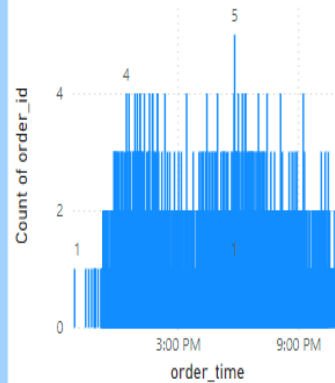


total quantity of each pizza category ordered

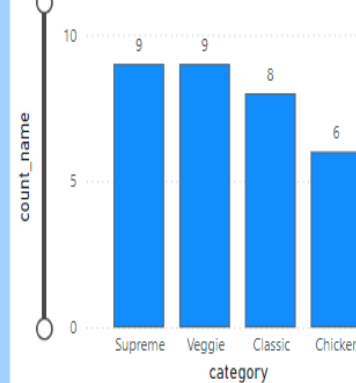




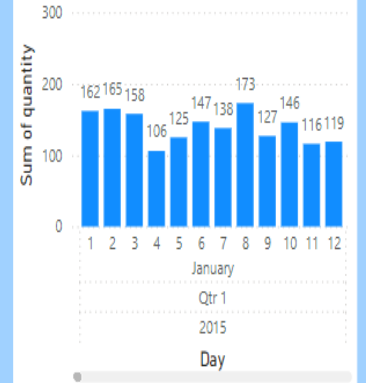
the distribution of orders by hour of the day.



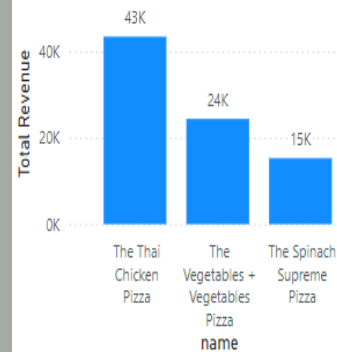
category-wise distribution of pizzas.



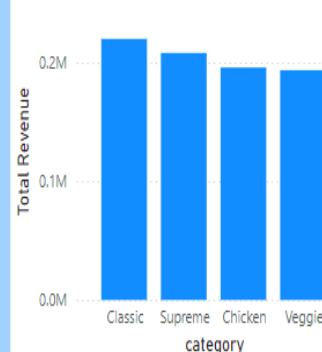
average number of pizzas ordered per day



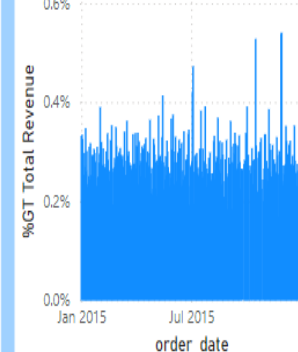
3 most ordered pizza types based on revenue



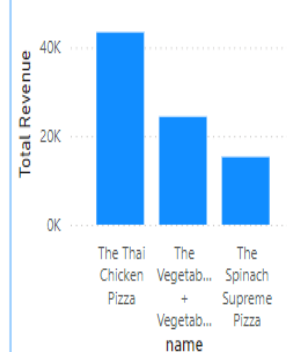
contribution of each pizza type to total revenue



cumulative revenue generated over time



3 most ordered pizza types based on revenue





Thank You

I Appreciate the opportunity to share
this project with you.