



Hello, my name is Priyal Keswani. In this project, I have solved SQL queries ranging from basic to advanced levels and provided answers to various questions related to pizza sales.

WHAT TABLES DATABASE CONTAIN

- 1) PIZZAS PIZZA_ID, PIZZA_TYPE_ID, SIZE, PRICE2) PIZZA_TYPE PIZZA_TYPE_ID, NAME,
- CATEGORY, INGREDIENTS
- **3) ORDERS -** ORDER_ID , ORDER_DATE , ORDER_TIME
- **4) ORDER_DETAILS -** ORDER_ID , ORDER_DETAILS_ID , PIZZA_ID , QUANTITY

QUESTIONS SOLVED

Basic:

- 1.Retrieve the total number of orders placed.
- 2. Calculate the total revenue generated from pizza sales.
- 3. Identify the highest-priced pizza.
- 4. Identify the most common pizza size ordered.
- 5.List the top 5 most ordered pizza types along with their quantities.

Intermediate:

- 1. Join the necessary tables to find the total quantity of each pizza category ordered.
- 2.Determine the distribution of orders by hour of the day.
- 3.Join relevant tables to find the category-wise distribution of pizzas.
- 4.Group the orders by date and calculate the average number of pizzas ordered per day.
- 5.Determine the top 3 most ordered pizza types based on revenue.

Advanced:

- 1.Calculate the percentage contribution of each pizza type to total revenue.
- 2. Analyze the cumulative revenue generated over time.
- 3.Determine the top 3 most ordered pizza types based on revenue for each pizza category.

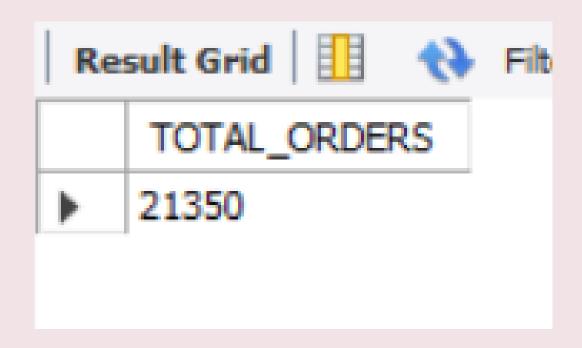
BASIC - 1)Retrieve the total number of orders placed.

```
SELECT

COUNT(ORDER_ID) AS TOTAL_ORDERS

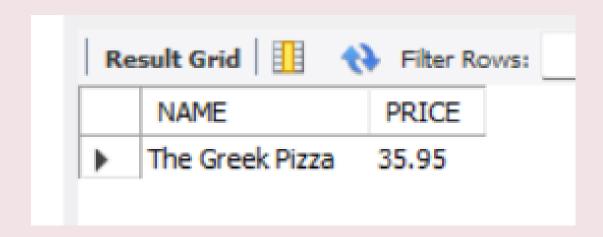
FROM

ORDERS;
```



BASIC- 2) Calculate the total revenue generated from pizza sales.

```
SELECT
    T.NAME, P.PRICE
FROM
    PIZZA_TYPES T
        JOIN
    PIZZAS P ON P.PIZZA_TYPE_ID = T.PIZZA_TYPE_ID
ORDER BY P.PRICE DESC
LIMIT 1;
```



BASIC-3)Identify the most common pizza size ordered.

```
P.SIZE, COUNT(D.ORDER_DETAILS_ID) AS ORDER_COUNT

FROM

PIZZAS P

JOIN

ORDER_DETAILS D ON P.PIZZA_ID = D.PIZZA_ID

GROUP BY P.SIZE

ORDER BY ORDER_COUNT DESC;
```

Re	sult Grid	Ⅲ ♦♦ Filb
	SIZE	ORDER_COUNT
•	L	18526
	M	15385
	S	14137
	XL	544
	XXL	28

BASIC-4) List the top 5 most ordered pizza types along with their quantities.

```
T.PIZZA_TYPE_ID , T.NAME, SUM(D.QUANTITY) AS QUANTITY

FROM

PIZZAS P

JOIN

PIZZA_TYPES T ON P.PIZZA_TYPE_ID = T.PIZZA_TYPE_ID

JOIN

ORDER_DETAILS D ON P.PIZZA_ID = D.PIZZA_ID

GROUP BY T.NAME , T.PIZZA_TYPE_ID

ORDER BY QUANTITY DESC

LIMIT 5;
```

Re	esult Grid 🎚 🐧	Filter Rows:	Export: V
	PIZZA_TYPE_ID	NAME	QUANTITY
•	classic_dlx	The Classic Deluxe Pizza	2453
	bbq_ckn	The Barbecue Chicken Pizza	2432
	hawaiian	The Hawaiian Pizza	2422
	pepperoni	The Pepperoni Pizza	2418
	thai_ckn	The Thai Chicken Pizza	2371
	-		

INTERMEDIATE-1) Join the necessary tables to find the total quantity of each pizza category ordered.

```
T.CATEGORY, SUM(D.QUANTITY) AS TOTAL_QTY

FROM

PIZZA_TYPES T

JOIN

PIZZAS P ON P.PIZZA_TYPE_ID = T.PIZZA_TYPE_ID

JOIN

ORDER_DETAILS D ON D.PIZZA_ID = P.PIZZA_ID

GROUP BY T.CATEGORY

ORDER BY TOTAL_QTY DESC;
```

Re	sult Grid 🛮 🔢	♦ Filter Rov
	CATEGORY	TOTAL_QTY
 	Classic	14888
	Supreme	11987
	Veggie	11649
	Chicken	11050

INTERMEDIATE-2) Determine the distribution of orders by hour of the day.

```
SELECT

HOUR(ORDER_TIME) AS HOUR, COUNT(ORDER_ID) AS COUNT_ORDER_ID

FROM

ORDERS

GROUP BY HOUR(ORDER_TIME)

ORDER BY COUNT_ORDER_ID DESC;
```

Re	sult Grid	Filter Row
	HOUR	COUNT_ORDER_ID
•	12	2520
	13	2455
	18	2399
	17	2336
	19	2009
	16	1920
	20	1642
	14	1472

INTERMEDIATE-3) Join relevant tables to find the category-wise distribution of pizzas.

```
SELECT

CATEGORY, COUNT(NAME)

FROM

PIZZA_TYPES

GROUP BY CATEGORY;
```

Re	sult Grid 🛚 🔢	Filter Rows:
	CATEGORY	COUNT(NAME)
•	Chicken	6
	Classic	8
	Supreme	9
	Veggie	9

INTERMEDIATE-4) Group the orders by date and calculate the average number of pizzas ordered per day.

```
SELECT

ROUND(AVG(QUANTITY),0) AS AVG_QTY

FROM

(SELECT

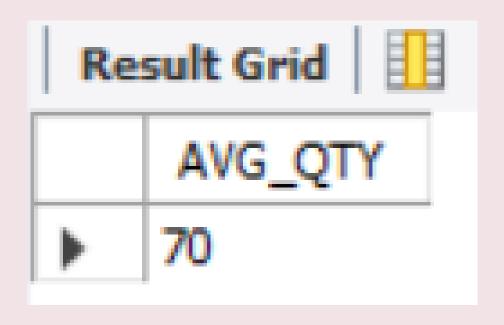
O.ORDER_DATE, SUM(D.QUANTITY) AS QUANTITY

FROM

ORDERS O

JOIN ORDER_DETAILS D ON O.ORDER_ID = D.ORDER_ID

GROUP BY O.ORDER_DATE , D.QUANTITY) AS ORDER_QUANTITY;
```



INTERMEDIATE-5) Determine the top 3 most ordered pizza types based on revenue.

```
SELECT
   T.PIZZA_TYPE_ID,
   T.NAME,
    SUM(D.QUANTITY * P.PRICE) AS REVENUE
FROM
    PIZZA TYPES T
        NIOL
    PIZZAS P ON P.PIZZA TYPE ID = T.PIZZA TYPE ID
        JOIN
    ORDER DETAILS D ON D.PIZZA_ID = P.PIZZA_ID
GROUP BY T.PIZZA_TYPE_ID , T.NAME
ORDER BY REVENUE DESC
LIMIT 3;
```

Re	Result Grid			Ę
	PIZZA_TYPE_ID	NAME	REVENUE	
•	thai_dkn	The Thai Chicken Pizza	43434.25	
	bbq_ckn	The Barbecue Chicken Pizza	42768	
	cali_ckn	The California Chicken Pizza	41409.5	

ADVANCED-1) Calculate the percentage contribution of each pizza type to total revenue.

```
SELECT
   T.CATEGORY,
   ROUND((SUM(D.QUANTITY * P.PRICE) / (SELECT
            ROUND(SUM(D.QUANTITY * P.PRICE), 2) AS TOTAL SALES
        FROM
            PIZZAS P
                JOIN
            ORDER_DETAILS D ON P.PIZZA_ID = D.PIZZA_ID)) * 100,2) AS REVENUE
FROM
    PIZZAS P
        JOIN
   PIZZA_TYPES T ON P.PIZZA_TYPE_ID = T.PIZZA_TYPE_ID
        JOIN
   ORDER DETAILS D ON D.PIZZA ID = P.PIZZA ID
GROUP BY T.CATEGORY
ORDER BY REVENUE DESC;
```

Re	sult Grid 🛮 🔢	★ Filter I
	CATEGORY	REVENUE
•	Classic	26.91
	Supreme	25.46
	Chicken	23.96
	Veggie	23.68

ADVANCE-2) Analyze the cumulative revenue generated over time.

```
SELECT ORDER_DATE , ROUND(SUM(REVENUE) OVER(ORDER BY ORDER_DATE),2) AS CUMULATIVE_REVENUE FROM

(SELECT O.ORDER_DATE ,
SUM(D.QUANTITY*P.PRICE) AS REVENUE FROM
ORDER_DETAILS D JOIN PIZZAS P ON D.PIZZA_ID = P.PIZZA_ID
JOIN ORDERS O ON O.ORDER_ID = D.ORDER_ID
GROUP BY O.ORDER_DATE ) AS SALES ;
```

	ORDER_DATE	CUMULATIVE_REVENUE
•	2015-01-01	2713.85
	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
	2045 24 22	04506 4

ADVANCE-3)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 RNN

FROM

(SELECT T.CATEGORY, T.NAME, SUM(D.QUANTITY*P.PRICE) AS REVENUE FROM

PIZZA_TYPES T JOIN PIZZAS P ON T.PIZZA_TYPE_ID = P.PIZZA_TYPE_ID

JOIN ORDERS_DETAILS D ON D.PIZZA_ID = P.PIZZA_ID

GROUP BY T.CATEGORY, T.NAME) AS A )

AS B WHERE RN<=3;
```

Result Grid			
	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	

Thank Jour